Enginyeria de Sistemes TIC

Projecte de curs de l'assignatura INFORMÀTICA

REDUCTIO

un minimitzador de funcions booleanes

Sebastià Vila-Marta

Departament de Disseny i
Programació de Sistemes Electrònics

Versió 3

Índex

1 Introducció

Aquest és el guió del projecte de curs de l'assignatura d'Informàtica de l'Enginyeria de Sistemes TIC de l'EPSEM. El projecte de curs és un treball en equip que té com a finalitat el disseny (parcial) i la implementació d'una aplicació informàtica realista en la que s'apliquen els coneixements i habilitats que s'han d'haver adquirit a l'assignatura.

En aquest apartat hi trobareu una descripció de l'objectiu d'aquest projecte i les condicions, organització i terminis en que cal dur-lo a terme.

1.1 Objectiu

L'objectiu d'aquest projecte és dissenyar i implementar una aplicació que llegeix una funció booleana, expressada com a suma de productes, i calcula una expressió minimal per a aquesta funció.

El projecte treballa amb conceptes principalment informàtics però també requereix certs coneixements matemàtics i de disseny de circuïts digitals.

Per tal que s'entengui bé què fa aquesta aplicació ho il·lustrarem amb un exemple. El circuït digital de la figura 1 té, com es pot veure, tres entrades anomenades a, b i c i una sortida anomenada y.

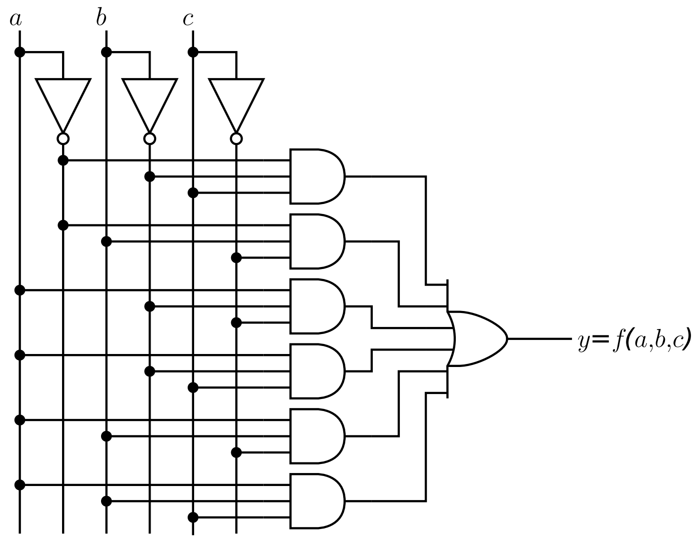

Figura 1: Circuït lògic no simplificat.

La tensió a la sortida f depèn dels valors de tensió a les entrades. És habitual denotar com un 0 els valors de tensió baixos i com un 1 els valors de tensió alts. Les components bàsiques del circuït, les portes, poden equiparar-se a una operació booleana elemental com, per exemple, la conjunció *i* — *and* en anglès. En aquest context, un circuït com l'anterior es pot modelar amb una funció booleana f definida:

$$f : \{0,1\}^3 \longrightarrow \{0,1\}$$

En el cas del circuït anterior, aquesta funció f es pot expressar així:

$$f(a,b,c) = \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + a\bar{b}c + ab\bar{c} + abc$$

on la suma denota l'operació o, el producte l'operació i i el barret denota el complement o negació. Fixeu-vos que la funció s'ha expressat com una suma de productes. Aquesta forma d'expressar la funció s'anomena *forma normal disjuntiva* (FND). La FND no és la única manera d'expressar la funció f. Si haguéssim escrit $f(a,b,c) = a + \bar{b}c + b\bar{c}$ ens estariem referint a la mateixa funció, això sí, escrita d'una altra forma.

Aquest fet no us hauria de sorprendre. Tots sabeu que, en el cas del les funcions reals, tant si es diu que $h(x) = (x+1)^2$ com si es diu que $h(x) = x^2 + 2x + 1$ s'està fent referència a la mateixa funció. Succeeix doncs que una cosa és la funció i una altra l'expressió algebraica que s'usa per escriure-la. Una mateixa funció pot ser escrita usant diverses expressions algebraiques.

Donada una funció booleana, la seva expressió algebraica ens indica com ha de ser el circuït digital que la implementa. Fixeu-vos en la funció f vista anteriorment. Si ens fixem en la seva expressió FND, el circuit que li correspon és el de la figura 1. En canvi, si ens fixem en l'altra expressió de la funció, $f(a,b,c) = a+\bar{b}c+b\bar{c}$, li correspondria el circuit de la figura 2. Existeix una corrèspondència entre l'expressió d'una funció booleana i el circuït que la implementa. Podem dir doncs que, per a una certa funció booleana g, existeix un conjunt de circuits diferents que la implementen. Notem com $E(g)$ el conjunt dels circuits equivalents que implementen la funció g.

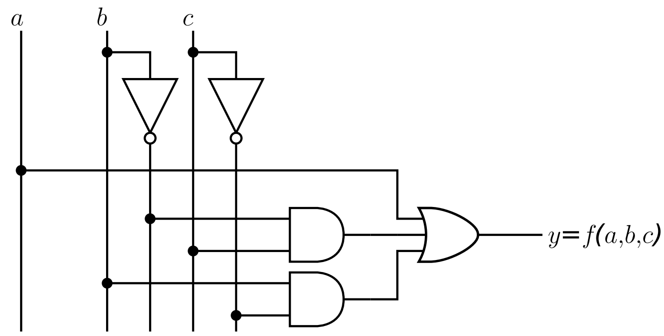

Figura 2: Circuït equivalent al de la figura 1. Observeu que és més senzill.

Donada una funció booleana g, els circuits d'$E(g)$ no són igual de complexos. Sempre n'hi haurà un o més que són més senzills que la resta: són els circuïts minimals d'$E(g)$. L'expressió algebraica que es correspon amb un circuït minimal, que també anomenarem *expressió minimal*, es distingeix per que té el menor nombre possible de conjuncions, disjuncions i negacions.

Ara podem precissar una mica més l'objectiu de l'aplicació que cal construïr. Es vol una aplicació tal que, donada una funció booleana expressada en FND, calculi una expressió minimal de la funció. Ja veieu que l'interès d'aquesta aplicació és gran: permet dissenyar circuits digitals combinatoris de la manera més barata possible!

2 Execució del projecte

2.1 Eines necessàries

Per dur a terme el projecte únicament són necessàries les eines habituals que s'ha usat durant el curs: l'editor emacs, l'interpret de Python i les seves llibreries i finalment l'eina per passar els tests nosetests.

2.2 Metodologia de treball

La metodologia que useu per fer aquest treball és una de les claus de l'èxit. Tingueu en compte els següents aspectes:

- Apliqueu les tècniques de TDD que ja coneixeu. Totes les funcions han d'anar acompanyades de la corresponent documentació i jocs de proves que permetin comprovar exhaustivament el programa.
- Treballar en equip no significa que tothom ha d'estar fent la mateixa feina simultàniament sinó just el contrari: cal separar la feina de forma que cada membre de l'equip pugui treballar pel seu compte i així avançar més depressa.
- La feina feta s'ha d'anar comprovant a mida que es fa. No podeu deixar les proves per al final del treball.
- És molt interessant que les persones de l'equip que NO han implementat un mòdul esmercin una mica de temps en llegir-lo, afegir-hi alguns casos de prova més i comprovar que funciona correctament. Això augmenta les possiblitats de detectar problemes en el mòdul.

2.3 Aspectes organitzatius

El projecte està dimensionat i dissenyat per a ésser treballat en equip. Cal que seguiu el següent esquema al treballar-hi:

1. Llegiu amb atenció individualment tot l'enunciat de punta a punta. Estudieu els coneixements previs.
2. Reuniu-vos i poseu en comú el que heu estudiat. Resoleu amb l'equip aquells punts que no heu entes.
3. Repartiu-vos la feina de les tasques 1,2 i 3. Fixeu-vos una data per acabar la feina.
4. Feu individualment les feines repartides.
5. Reuniu-vos i poseu la feina feta en comú. Resoleu allò que no hagi quedat clar.
6. Repartiu-vos la implementació dels mòduls `terme` i `ct`. Fixeu-vos una data per acabar la feina.
7. Implementeu i testejeu individualment el mòdul que us ha tocat fins que el tingueu a punt.
8. Reuniu-vos, intercanvieu-vos la feina i poseu en comú el resultat. Estudieu com atacar el mòdul `minim`. Fixeu-vos una data per acabar la feina.
9. Individualment estudieu com implementar el mòdul `minim`. Feu proves, implementeu-ne parts, etc. Fixeu-vos una data per acabar la feina.
10. Reuniu-vos i implementeu de manera conjunta els mòduls `minim` i `reductio`. Dissenyeu els tests convenients i comproveu que passen sense problemes.
11. Repartiu-vos la tasca de documentar el projecte.

2.4 Lliurables

1. Manual d'usuari. Un petit document en que s'explica a un usuari final per que serveix i com cal usar el programa que heu implementat.
2. Memòria tècnica: codi font del projecte documentat convenientment.
3. Memòria d'execució: Cal que entregueu una taula que reflecteixi l'execució del projecte. Aquesta taula ha de tenir quatre columnes: una per la data i una per a cada persona que forma l'equip. En aquesta taula hi consignareu una casella per cada dia que treballeu en el projecte. En aquesta casella indicareu molt resumidamenbt què heu fet i quant temps hi heu dedicat. Finalment sumareu tots els temps dedicats. La taula ha de semblar aquesta:

Data	Pep	Maria	Aina
2/12		Llegir enunciat (2h)	Llegir enunciat (2h) Pensar exemple (1h)
3/12	Reunió treball (1h)	Reunió treball (1h)	Reunió treball (1h)
	1h feina	3h feina	4h feina

3 Coneixements previs

En aquest apartat s'expliquen els coneixements previs necessaris per entendre i desenvolupar l'aplicació que ens ocupa.

3.1 Funcions, formes normals i minterms

Direm que una funció booleana està expressada en *forma normal disjuntiva.* abreviada FND, ssi és una disjunció d'un conjunt de conjuncions en les que cada variable ocorre com a màxim una vegada. Per exemple, les següents expressions de la funció f són FDN:

$$\begin{aligned} f(a,b,c) &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + a\bar{b}c + ab\bar{c} + abc \\ f(a,b,c) &= a + \bar{b}c + b\bar{c} \end{aligned}$$

Una funció booleana s'expressa en *forma normal disjuntiva plena*, abreviat FNDP, ssi és FND i a més a cada conjunció apareixen totes les variables de la funció. Per exemple, en la funció anterior només la primera expressió és FNDP. Els termes conjuntius d'una funció FNDP sovint s'anomenen *minterms*. Així, en l'exemple anterior, $\bar{a}\bar{b}c$ és un minterm.

Tota funció booleana es pot tabular indicant per a cada valor de les variables el valor que pren la funció. D'aquesta taula en diem *taula de veritat*. En el cas d'f la taula de veritat és la que segueix:

	Variables			
n	a	b	c	$f(a,b,c)$
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Com que cada combinació de valors de les variables pot ser interpretada com un número natural escrit en base 2, és fàcil establir una numeració de tots els possibles valors d'entrada per a una funció booleana. Per exemple, en la funció f al valor d'entrada que correspon a $a = 0$, $b = 1$ i $c = 1$ li correspon el número 3 atès que:

$$\mathbf{0} \cdot 2^2 + \mathbf{1} \cdot 2^1 + \mathbf{1} \cdot 2^0 = 3$$

En la taula anterior, la columna etiquetada n fa referència a aquesta numeració.

Entre la taula de veritat i la FNDP de la funció corresponent hi ha moltes «coincidències». La primera que cal observar és la relació entre els minterms de la FNDP i les files de la taula. Fixeu-vos que a cada minterm d'f li correspon una fila de la taula de veritat. Simplement cal substituir en el minterm les variables negades per 0 i les altres per 1. Per exemple, al minterm $\bar{a}b\bar{c}$ li correspon la fila amb valors 010. Quan un minterm l'escrivim usant els noms de les variables, com per exemple $\bar{a}b\bar{c}$, direm que té *forma algebraica*. D'altra banda quan un minterm l'escrivim usant 0's i 1's direm que té *forma binària*. Fixeu-vos que les files de la taula de veritat que corresponen als minterms d'f són precissament aquelles en les que la funció val 1.

Tasca 1
Per què les files corresponents als minterms de la FNDP són els que corresponen a valors 1 de la funció? Per qualsevol f això és així?

Aquesta coincidència i el fet que cada a cada fila de la taula de veritat li correspon un enter que la identifica permet un pas sorprenent: expressar una funció booleana com un conjunt de números naturals. En efecte, al minterm $\bar{a}b\bar{c}$ li correspon la fila amb valors 010 i a aquesta fila el natural $n = 2$. Per tant parlar del minterm $\bar{a}b\bar{c}$ o del minterm 2 és exactament el mateix. Com una funció en FNDP no és altra cosa que un conjunt de minterms també podem pensar que és un conjunt d'enters. Així, podem denotar la funció f com:

$$f(a,b,c) = \{1,2,4,5,6,7\}$$

3.2 Minimització de Quine-McCluskey

Per a minimitzar la funció usarem l'algoritme de Quine-McCluskey, dissenyat per Edward J. McCluskey el 1956. Aquest apartat presenta una sèrie de conceptes importants per a entendre aquest algoritme així com una descripció intuitiva del seu funcionament. La seva comprensió ens permetrà seguir el projecte.

Tasca 2
Quina és la referència bibliogràfica del document en que es descriu per primera vegada l'algoritme de Quine i MacCluskey? Qui són Quine i McCluskey?

3.2.1 Implicants primers i minimització

Noteu que, en una funció booleana en FNDP a vegades és possible substituir dos minterms per una única conjunció. Per exemple, en la funció f que hem estat usant com a exemple, els minterms $a\bar{b}\bar{c}$ i $a\bar{b}c$ es podrien substituir, aplicant les regles de l'àlgebra de Boole, per un únic terme $a\bar{b}$ atès que $a\bar{b}\bar{c} + a\bar{b}c = a\bar{b}(\bar{c} + c) = a\bar{b} \cdot 1 = a\bar{b}$. Si ho fessim podríem escriure:

$$\begin{aligned} f(a,b,c) &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + a\bar{b}c + ab\bar{c} + abc \\ &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b} + ab\bar{c} + abc \end{aligned}$$

Fixeu-vos que la fórmula resultant és més senzilla que l'original i que, tot i haver perdut la condició de FNDP, continua essent una FND. D'aquestes conjuncions que poden *representar* més d'un minterm simultàniament com és el cas de $a\bar{b}$ en diem *implicants.* Podem expressar el fet que l'implicant $a\bar{b}$ representa simultàniament als minterms $a\bar{b}\bar{c}$ i $a\bar{b}c$ amb un graf com el de la figura 3.

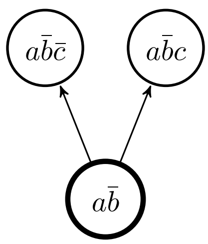

Figura 3: A la funció f, l'implicant $a\bar{b}$ representa als minterms $a\bar{b}\bar{c}$ i $a\bar{b}c$ ja que $a\bar{b}\bar{c} + a\bar{b}c = a\bar{b}$.

Si juguem una mica amb la fórmula descobrirem altres implicants, per exemple podem substituïr $ab\bar{c}$ i abc per ab i deixar la fórmula d'f com:

$$\begin{aligned} f(a,b,c) &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + a\bar{b}c + ab\bar{c} + abc \\ &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b} + ab\bar{c} + abc \\ &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b} + ab \end{aligned}$$

Però ara, si sou observadors, haureu notat que podem substiuïr els dos implicants primers anteriors, $a\bar{b}$ i ab, per un nou implicant més simple encara: a. L'implicant a representa a 4 minterms de la funció f. Si apliquem aquesta substitució a la fórmula de f tenim una expressió encara més simple que l'anterior:

$$\begin{aligned} f(a,b,c) &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + a\bar{b}c + ab\bar{c} + abc \\ &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b} + ab\bar{c} + abc \\ &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b} + ab \\ &= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a \end{aligned}$$

En general, dos minterms o dos implicants qualsevols u i w poden ser representats per un nou implicant sii es donen les dues condicions següents:

1. u i w tenen exactament les mateixes variables.
2. Existeix una i només una variable de u que surt complementada a w.

La idea de l'algoritme de minimització de Quine-McCluskey es basa calcular tots els possibles implicants d'una funció en FNDP i després, en un segon pas, determinar la mínima quantitat possible d'implicants que permetin descriure la funció.

Per a la funció f el conjunt d'implicants primers el podem veure en el graf de la figura 4. En aquest graf, els implicants estàn encerclats per un cercle gruixut, els minterms per un de prim i les arestes indiquen a quins minterms o implicants representa cada implicant. Noteu que alguns dels implicants són primers, com per exemple a, mentre que altres no, com és el cas d'$a\bar{c}$.

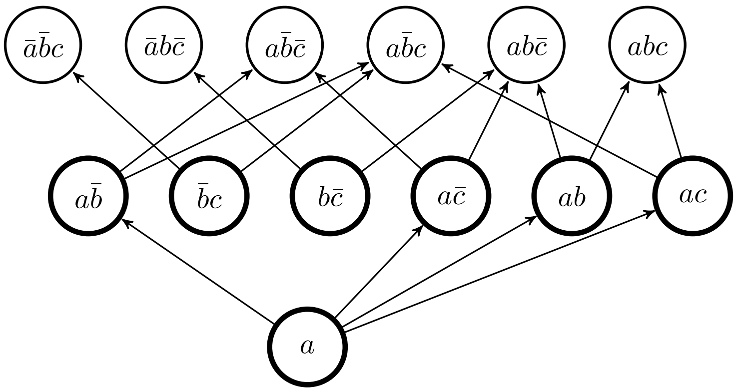

Figura 4: Graf dels implicants d'f, remarcats en gruixut.

A la figura 5 s'han triat alguns implicants primers, indicats en vermell, que en conjunt representen a tots els minterms de la funció, indicats en groc. Això ens indica que podem expressar la funció f amb la següent fórmula en FND:

$$f(a, b, c) = a + \bar{b}c + b\bar{c} \tag{1}$$

És important adonar-se'n de tres fets:

1. Només cal tenir en compte aquells implicants que no estan representats per altres implicants. Aquests implicants els anomenem *implicants primers*. A la funció f, per exemple, l'implicant ab no cal tenir-lo en compte ja que es troba representat per a. a és un implicant primer.
2. Alguns implicants primers cal triar-los obligatòriament. En efecte, si observeu el graf de la figura 5 veureu que el minterm $\bar{a}\bar{b}c$ està representat per un únic implicant, el $\bar{b}c$. Per tant aquest implicant s'ha de triar obligatòriament. D'aquests implicants primers en diem *implicants essencials*.
3. Hi poden haver mimterms que no són representats per cap implicant. En aquest cas, formaràn part de la fórmula final obligatòriament.

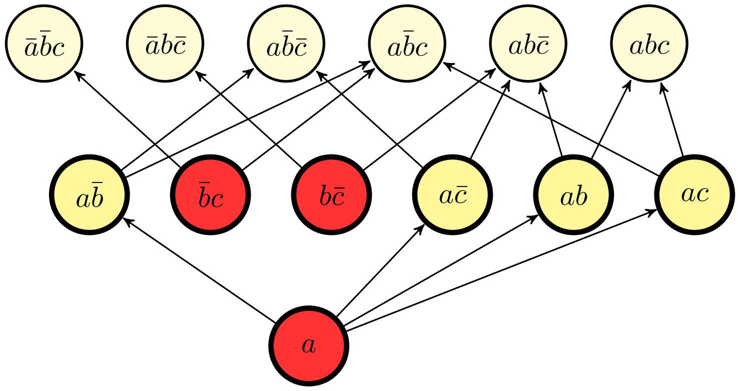

Figura 5: En vermell els implicants que minimitzen la funció.

Tasca 3
Usant diagrames d'Euler-Venn expresseu les relacions que existeixen entre els següents conceptes que s'usen en aquest enunciat: terme, minterm, implicant, implicant primer, implicant essencial.

Com no hi ha cap conjunt més petit d'implicants que pugui representar tots els minterms d'f, la fórmula 1 és minimal. Així doncs hem passat d'una funció f en FNDP a una funció f en FND i minimal com volíem. La idea bàsica de l'algoritme de Quine-McCluskey consisteix a considerar la llista de minterms de la funció FNDP i calcular la funció minimal fent:

1. Afegeix a la funció minimal els minterms sense representant i els cancela de la llista de minterms.
2. Afegeix a la funció minimal els implicants primers essencials i cancela els minterms representats.
3. Tria l'implicant primer que més minterms cancela, l'afegeix a la funció minimal i cancela tots els minterms a qui representa.
4. Repeteix el pas anterior fins que no queden minterms a la llista.

3.2.2 Una altra perspectiva dels implicants

El procés de simplificació que hem vist abans el podem estudiar ara considerant la forma binària dels minterms en comptes de la seva forma algebraica. Així, ara no parlarem del minterm $\bar{a}b\bar{c}$ sinó del minterm 010.

Des d'aquesta perspectiva, dos minterms m_1 i m_2 poden ser representats per un implicant primer sii el nombre de 1's de $\overline{m_1 \oplus m_2}$ és exactament 1, és a dir, si m_1 i m_2 únicament difereixen en una de les variables[1]. Per exemple, els minterms 010 i 011 difereixen només en el bit de menys pes i, per tant, poden ser representats per un únic implicant primer. Aquest implicant el representarem com 012. Fixeu-vos que simplement substituïm el «bit diferent» per un 2.

[1] $\oplus$ fa referència a l'o exclusiva

En realitat el que hem fet en el paràgraf anterior no és res nou. Fixeu-vos que suceeix si ho traslladem a la forma algebraica:

$$\begin{aligned} 010 &\longrightarrow \bar{a}b\bar{c} \\ 011 &\longrightarrow \bar{a}bc \\ 012 &\longrightarrow \bar{a}b \end{aligned}$$

Noteu com el 2 en l'implicant significa absència de variable en la forma algebraica. La figura 6 mostra el graf d'implicants de la funció f afegint també la forma binària entre parèntesis. Fixeu-vos que, per determinar si donats dos implicants en forma binària existeix un implicant més senzill que els representi, cal tractar el dígit 2 tal i com ho fem amb el 1 o el 0. Així, per exemple, els implicants 102 i 112 només difereixen en un dígit, el del mig, i per tant es poden substituir pel 122 tal com mostra el graf. En canvi, els implicants 112 i 121 es diferencien en dos dígits i, per tant, no es poden substituïr per un tercer.

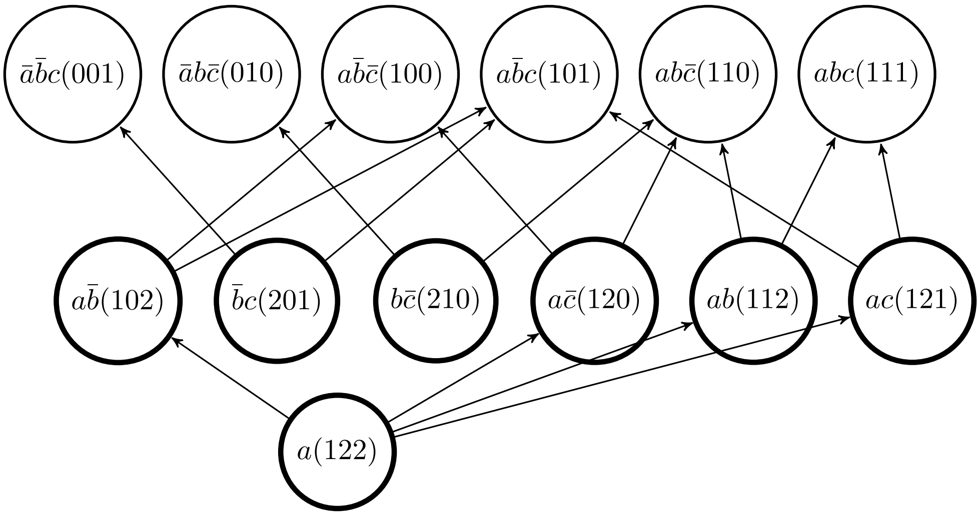

Figura 6: Graf dels implicants primers d'f ara també en forma binària.

Tasca 4
Trieu tres funcions boolenes, u, v i w, de 3, 4 i 5 paràmetres respectivament. Expresseu-les en FNDP. Feu el graf d'implicants (com el de la figura 4) i trobeu-ne un conjunt mínim. Expresseu les funcions en FND minimal.

Aquests exemples us seràn útils per poder fer jocs de proves interessants en el projecte.

4 Estructura de mòduls

El projecte està composat de diversos mòduls. Cada mòdul contindrà una part de la funcionalitat. D'aquesta manera l'organització del treball, les proves i el reaprofitament se'n veuen beneficiats.

En aquest apartat es mostra amb un diagrama, vegeu la figura 7 quins mòduls composen el projecte i com estan organitzats. Aquesta informació és important per tenir una visió global de la seva arquitectura.

Cada mòdul correspon a un fitxer Python que conté un conjunt de variables i/o funcions relacionades. Una fletxa $A \rightarrow B$ significa que el mòdul B utilitza el mòdul A, per tant el mòdul B té un **import** del mòdul A.

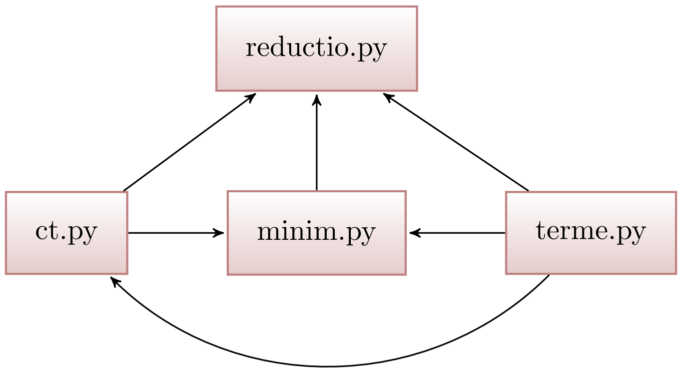

Figura 7: Diagrama de mòduls i dependències del projecte.

La funcionalitat de cada mòdul es resumeix a continuació:

ct Aquest mòdul ofereix operacions per treballar amb conjunts de termes.

fbool És el mòdul responsable de representar i oferir operacions per treballar amb funcions booleanes.

minim És el mòdul que conté la funcionalitat associada a la minimització pròpiament.

reductio És el mòdul principal. L'encarregat de la interfície de comandes. L'usuari interacciona amb aquest mòdul.

5 El mòdul terme

Un *terme* en el context d'aquest projecte és un objecte Python que tant pot representar un minterm com un implicant. Un terme es representarà com una tupla d'idèntica longitud que el nombre de variables de la funció a minimitzar i amb components que prenen valors en el conjunt $\{0, 1, 2\}$. D'una forma més rigorosa podriem dir que un terme es representa per un $x \in \{0, 1, 2\}^n$ essent n el nombre de variables de la funció a minimitzar.

El primer que implementarem són les funcions de conversió entre cadenes i termes:

Tasca 5
Implementeu les següents funcions del mòdul terme:

1. *str_a_terme(s) Donada un cadena `s` formada únicament pels caràcters `'0'`, `'1'` i `'2'`, retorna el terme que li correspon. Per exemple, donada la cadena `'01002'` retorna el terme (0,1,0,0,2).*
2. *terme_to_str(t) Donat un terme `t`, retorna la cadena que li correspon.*

A continuació podeu implementar un seguit d'operacions sobre termes que més endavant seràn útils. Per implementar-los és convenient que tingueu clar el concepte de «paràmetre opcional» que trobareu explicat en l'apartat 7.10[2] del llibre de l'assignatura.

Tasca 6

1. *compta(t, d=1) Retorna el nombre de dígits amb valor d que conté el terme t. El paràmetre d és opcional i el seu valor per omissió és 1.*
2. *es_minterm(t) Retorna True sii el terme t representa un minterm.*
3. *potencia(t) Retorna en nombre de digits 2 del terme t.*
4. *son_reduibles(t1, t2) Retorna True sii els termes t1 i t2 es poden reduir a un implicant, és a dir, si t1 i t2 es diferencien només en un dígit.*
5. *implicant(t1, t2) Retorna l'implicant dels termes t1 i t2. S'assumeix que t1 i t2 tenen un implicant, altrament el resultat de l'operació no està definit.*
6. *representa(i, t) Retorna True sii l'implicant i representa el terme t.*

6 El mòdul ct

Aquest mòdul té com a objectiu oferir la funcionalitat necessària per treballar amb conjunts de termes. Un conjunt de termes és una col·lecció de termes del mateix nombre de variables en la que cap terme hi és més d'una vegada. El llenguatge Python té predefinit un tipus de dades (set) amb aquesta funcionalitat però aquí no en farem ús.

Per repesentar un conjunt de termes usarem una llista. Així, per exemple, el conjunt de minterms $\{010, 110, 201\}$ es representarà com [(0,1,0), (1,1,0), (2,0,1)]. Les operacions procuraràn garantir que mai existeixen termes repetits dins la llista i, per tant, es comporta com un conjunt. Fixeu-vos que els conjunt ho són de termes i, per tant, poden contenir tant minterms com implicants.

A continuació implemenarem algunes funcions sobre conjunt que seràn utils més endavant:

Tasca 7

Implementeu les funcions següents:

1. *buit() Retorna un conjunt de termes buit.*
2. *afegeix(c,t) Retorna el conjunt resultant d'afegir al conjunt c el terme t. Naturalment si el conjunt ja contenia el terme no l'afegeix dues vegades. Ha de comprovar que el terme afegit té la mateixa dimensió que els termes existents.*
3. *unio(c1, c2) Retorna el conjunt $c1 \cup c2$. La unió ha tenir present que el resultat no pot contenir termes repetits.*
4. *diferencia(c1, c2) Retorna el conjunt $c1 - c2$.*

[2] http://ocw.itic.cat/assignatures/inf/Apunts/introduccio-a-la-programacio

5. *nvars(c) Retorna el nombre de variables dels termes que conté el conjunt.*
6. *clona(c) Retorna un conjunt clon de c. És a dir conté els mateixos elements però és un conjunt diferent.*
7. *ct_a_str(c) Retorna un cadena que representa la funció caracteritzada pels termes del conjunt c. Per exemple, per a la funció representada pel conjunt $\{101, 200\}$, la cadena que retorna és* `"f(A,B,C) = AB'C + B'C'"`*. Les variables sempre s'anomenen amb majúscules i començant per la* `'A'`*. Els literals negats s'escriuen amb un apostrof darrera de la variable. Així, aquesta operació escriu $\bar{a}$ com* `"A'"`*. Fixeu-vos que en aquest cas, l'operació no retorna sempre minterms sinó que s'escriu tal i com indica la representació.*

7 El mòdul minim

Aquest mòdul conté el cor del minimitzador de funcions booleanes. Per implementar-lo seguiu la seqüència de tasques següents:

Tasca 8
Implementeu la funció implicants(f). Aquesta funció retorna el conjunt de tots els implicants primers possibles de f. Podeu assumir que la representació de f està en FNDP. Procediu de la següent forma, que és matussera però senzilla:

Sigui $M_0 = \{m_0, m_1, \ldots, m_k\}$ el conjunt de minterms que representa la funció f. Anem a construïr un nou conjunt M_1. A tal efecte recorreu totes les possibles parelles de termes de M_0, és a dir totes les parelles $(m_i, m_j) \in M_0 \times M_0$ amb $i < j$. Si la parella en qüestió té un implicant que la representa, afegiu-lo al conjunt M_1. Ara, seguint el mateix procediment, calculeu M_2 a partir dels termes de M_1 i així successivament fins que arribeu a un conjunt M_p en el que cap parella de termes pot ser representada per un implicant. Haureu obtingut una seqüència de conjunts $M_0, M_1, M_2, \ldots, M_p$. Ara només cal que la funció retorni el conjunt I definit com:

$$I = \bigcup_{i=1}^{p} M_i$$

Si considereu la funció booleana f que hem usat d'exemple en tot aquest document, el conjunt d'implicants que resultaria de la tasca anterior coincideix amb els que, en l'exemple de la figura 4, estan encerclats en gruixut. Si us hi fixeu, el el conjunt d'implicants hi ha redundància: per exemple, l'implicant 102 ja està representat pel 122, que també és en el conjunt. En certa manera el 102 sobra. El següent elimina aquests implicants redundants i retorna els implicants primers..

Tasca 9
Implementeu la funció primers(c) que, donat un conjunt d'implicants c retorna el conjunt obtingut en eliminar els implicants redundants de c.

A tal efecte seguiu aquesta estratègia: creeu un clon del conjunt c*, diguem-ne* r*. Recorreu tots i cadascun dels implicants de* c *i, per a cada implicant* i*, elimineu del conjunt* r *tots aquells elements que estan representats per* i*. En finalitzar el recorregut* r *conté els implicants sense redundància, o sia els* implicants primers.

La següent tasca permet calcular un conjunt d'implicants primers mínim donat un conjunt de minterms.

Tasca 10
Implementeu la funció termes_minimals(m) *que retorna un conjunt minimal de termes que representen a* m*, essent* m *un conjunt de minterms i* p*. Per implementar aquesta funció procediu amb els següents passos:*

1. *Sigui* M *un conjunt de minterms i* P *el conjunt d'implicants primers d'*M*. Sigui* R *un conjunt de termes inicialment buit.*
2. *Recorreu els minterms de* M *i per a cada minterm* $m \in M$ *que no es representat per cap implicant primer de* P*, afegiu* m *a* R *i esborreu* m *de* M*.*
3. *Recorreu els minterms que queden a* M *i per cada minterm* $m \in M$ *que només tingui un implicant primer* $p \in P$ *que el representi, afegiu* p *a* R *i esborreu de* M *tots el minterms representats per* p*.*
4. *Si a* M *encara queda algun minterm* m*, busqueu a* $p \in P$ *l'implicant primer que representa a* m *i que té el màxim nombre de 2's. Afegiu* p *a* R *i esborreu d'*M *tots els minterms representats per* p*.*
5. *Repetiu el pas anterior fins que no quedin minterms a* M*.*

En acabar, R *és el conjunt de termes que es volia calcular.*

Finalment ja només queda lligar-ho tot plegat. Això és l'objectiu de la següent tasca:

Tasca 11
Implementeu la funció minimitza(c)*. Aquesta funció té com a paràmetre una funció booleana expressada en FNDP com un conjunt de termes* c *i retorna una funció booleana equivalent però minimal expressada, també, com a conjunt de termes.*

8 El mòdul reductio

Aquest mòdul es el que interacciona amb l'usuari. En essència ofereix un programa que llegeix una funció booleana continguda en un fitxer de text i escriu a la terminal una expressió minimal per a la mateixa funció. La funció d'entrada s'assumeix que està en FNDP i es llegirà com una taula de minterms. El format d'aquesta taula està orientat a línies. Cada línia representa un minterm. Així, la funció f que ens ha servit d'exemple en aquest enunciat es descriuria com:

```
0011
0101
1001
1011
1101
1111
```

La funció minimitzada s'escriurà en notació convencional:

```
f(A,B,C) = A + B'C + BC'
```

La tasca corresponent és doncs:

Tasca 12
Escriviu el programa principal en aquest mòdul. Aquest programa s'ha d'invocar indicant en la mateixa línia d'ordres el nom del fitxer que conté la funció a minimitzar. El fitxer conté una seqüència de minterms com s'ha dit anteriorment. El programa calcula una expressió minimal i l'escriu a la terminal.