fcfm
Ingeniería
FACULTAD DE CIENCIAS
FÍSICAS Y MATEMÁTICAS
UNIVERSIDAD DE CHILE

Tarea 1

CC3101: Discrete Mathematics for Computer Science 2019

Submitted To:
Pablo Barcelo

Submitted By :
Alexandre Poupeau

Exercice 1

For the rest of the exercice 1, we assume that :

$p \in CNF \iff \exists k \in \mathbb{N}$ and $\forall i \in [\![1, k]\!], \exists J_i \in \mathbb{N}$ such that $p = \bigwedge_{1 \le i \le k} C_i = \bigwedge_{1 \le i \le k} (\bigvee_{1 \le j \le J_i} l_{i,j})$ where $l_{i,j}$ is a literal. C_i is the symbol we will used to represent clauses.

1.1) Prove that all logical propositional formulas can be expressed in CNF. Before starting anything we will consider the following property as true since it is trivial. There is no need to make a demonstration for it.

Property 1. $(\Phi, \Omega) \in CNF \implies (\Phi \wedge \Omega) \in CNF$

In order to demonstrate what we want to demonstrate we will prove two properties first. Here is the first property :
Property 2. P_n : *If* $(\Phi, \Omega) \in CNF$ *and are especially of the following form :* $\Phi = \bigwedge_{1 \le i \le k} C_i$ *and* $\Omega = \bigwedge_{1 \le i \le k'} C'_i$ *with* $k, k' \le n \implies (\Phi \vee \Omega) \in CNF$

Proof. Case $n = 1$:

$$\begin{aligned}
\Phi \vee \Omega &= \bigvee_{1 \le j \le J_1^\Phi} l_{1,j}^\Phi \vee \bigvee_{1 \le j \le J_1^\Omega} l_{1,j}^\Omega \\
&= \bigwedge_{1 \le i \le k} (\bigvee_{1 \le j \le J'_i} l'_{i,j})
\end{aligned}$$

Where $k = 1$, $J'_1 = J_1^\Phi + J_1^\Omega$ and $l'_{1,j} = \begin{cases} l_{1,j}^\Phi \ , \forall j \in [\![1, J_1^\Phi]\!] \\ l_{1,j-J_1^\Phi}^\Omega , \forall j \in [\![J_1^\Phi + 1, J_1^\Phi + J_1^\Omega]\!] \end{cases}$

Then for $n = 1$, the property is true. Now let us suppose that P_n is true and let us prove P_{n+1}. Let us consider the hard case directly, where $\Phi = \bigwedge_{1 \le i \le n+1} C_i$ and $\Omega = \bigwedge_{1 \le i \le n+1} C'_i$.

$$\begin{aligned}
\Phi \vee \Omega &= \bigwedge_{1 \le i \le n+1} C_i \vee \bigwedge_{1 \le i \le n+1} C'_i \\
&= (\bigwedge_{1 \le i \le n} C_i \wedge C_{n+1}) \vee (\bigwedge_{1 \le i \le n} C'_i \wedge C'_{n+1}) \\
&= (\Phi' \wedge C_{n+1}) \vee (\Omega' \wedge C'_{n+1}) \\
&= (\Phi' \vee \Omega') \wedge (C_{n+1} \vee \Omega') \wedge (\Phi' \vee C'_{n+1}) \wedge (C_{n+1} \vee C'_{n+1})
\end{aligned}$$

Because of the induction hypothesis we have that $(\Phi' \vee \Omega')$, $(C_{n+1} \vee \Omega')$, $\Phi' \vee C'_{n+1})$ and $(C_{n+1} \vee C'_{n+1})$ are in CNF. Moreover, thanks to the Property 1, we have $(\Phi \vee \Omega) \in CNF$.

If Φ and Ω are not in this form (example: $\Phi = \bigwedge_{1 \leq i \leq n+1} C_i$ and $\Omega = \bigwedge_{1 \leq i \leq n-5} C'_i$ or $\Phi = \bigwedge_{1 \leq i \leq n-2} C_i$ and $\Omega = \bigwedge_{1 \leq i \leq n-3} C'_i$), the result is even easier to prove.

Thus the property is true $\forall n \in \mathbb{N}^*$.

$\square$

The second property we want to prove is the following one :

Property 3. $P_n : \Phi = \bigwedge_{1 \leq i \leq n} C_i \in CNF \implies (\neg\Phi) \in CNF$

Proof. Case $n = 1$: $\Phi = \bigvee_{1 \leq j \leq J_1} l_{1,j}$ then :

$$\begin{aligned}(\neg\Phi) &= \bigwedge_{1 \leq j \leq J_1} (\neg l_{1,j}) \\ &= \bigwedge_{1 \leq i \leq k} \Big(\bigvee_{1 \leq j \leq J'_i} l'_{i,j} \Big)\end{aligned}$$

Where $k = J_1$, $\forall i \in [\![0, k]\!], J'_i = 1$ and $l'_{i,j} = \neg l_{j,i}$.

Case $n + 1$: Let us suppose P_n true and prove P_{n+1}.

$$\begin{aligned}(\neg\Phi) = \neg\Big(\bigwedge_{1 \leq i \leq n+1} C_i\Big) &= \neg\Big(\bigwedge_{1 \leq i \leq n} C_i \wedge C_{n+1}\Big) \\ &= \neg(\Phi' \wedge C_{n+1}) \\ &= (\neg\Phi') \vee (\neg C_{n+1})\end{aligned}$$

$(\neg\Phi') \in CNF$ by induction hypothesis and $(\neg C_{n+1}) \in CNF$ (this is exactly like the case $n = 1$). Thanks to the Property 2 ($(\Phi, \Omega) \in CNF \implies (\Phi \vee \Omega) \in CNF$), we finally have that $(\neg\Phi) \in CNF$.

Thus the property is true $\forall n \in \mathbb{N}^*$.

$\square$

Now we are going to prove the result we want by induction on the language $\mathcal{L}(\mathcal{P})$.

Proof. Base case : $p \in \mathcal{P}$. Obviously $p \in CNF$ because $p = \bigwedge_{1 \leq i \leq k} C_i$ where $k = 1$ and $C_1 = p$.

Induction case : Let us suppose $(\Phi, \Omega) \in CNF$.

- $(\Phi \wedge \Omega) \in CNF$ (Property 1 trivial)
- $(\Phi \vee \Omega) \in CNF$ (Property 2)
- $(\neg\Phi) \in CNF$ (Property 3)
- $(\Phi \rightarrow \Omega) \in CNF$ (by combinaison of Prop 2 and 3 because $(\Phi \rightarrow \Omega) \equiv ((\neg\Phi) \vee \Omega)$)

□

We finally proved that all formulas in $\mathcal{L}(\mathcal{P})$ can be expressed in CNF.

1.2) Prove that for all formulas in CNF of n clauses, there exists a valuation where at least $n/2$ clauses are satisfied. First we need to rewrite the problem in a more mathematical form :

$$\forall \alpha \in CNF \text{ where } \alpha = \bigwedge_{1 \leq i \leq n} C_i \text{ with } n \in \mathbb{N}^*, \exists \sigma : \mathcal{P} \rightarrow \{0,1\} \text{ such that } \sum_{k=1}^{n} \sigma(C_k) \geq \frac{n}{2}$$

We need to define how we construct clauses in this question. Let $\mathcal{L}^{\mathcal{C}}(\mathcal{P})$ be the language that defines the clauses. It is defined by induction :

Base : If $p \in \mathcal{P} \implies p \in \mathcal{L}^{\mathcal{C}}(\mathcal{P})$ and $(\neg p) \in \mathcal{L}^{\mathcal{C}}(\mathcal{P})$.

Induction : If $\alpha \in \mathcal{L}^{\mathcal{C}}(\mathcal{P})$ and $p \in \mathcal{P}$, then $(p \vee \alpha) \in \mathcal{L}^{\mathcal{C}}(\mathcal{P})$ and $((\neg p) \vee \alpha) \in \mathcal{L}^{\mathcal{C}}(\mathcal{P})$.

We will first prove a very intuitive property :

Property 4. *If C is a clause and T its truth table composed of n_T valuations $\implies$ At least $\frac{n_T}{2}$ valuations satisfy C.*

Let us prove the previous property :

Proof. Base case : C clause is composed of only one literal l. l can be p or $(\neg p)$ of a $p \in \mathcal{P}$. In both cases, it is trivial that there exists 1 valuation (over 2) that satifies C.

Inductive case : Let $C \in \mathcal{L}^{\mathcal{C}}(\mathcal{P})$ where half of the valuations satisfy C, and l be a literal.

- Case l literal already in C. Then $(l \vee C) \equiv C$. This case is trivial and the result holds by induction hypothesis.
- Case l not already in C :
  - Case $l = (\neg p)$ and p already in C (or the contrary without loss of generality). In this case, all the valuations satisfy $(l \vee C)$, which is obviously more than half.
  - Case l is a new proposition not in C (or the negation of a new proposition without loss of generality) :

l	C	$(l \vee C)$
0	0	0
0	1	1
1	0	1
1	1	1

$2n_T$ valuations are available here because n_T were available for C. We can easily deduce that at least n_T valuations satisfy $(l \vee C))$. simply because of l.

□

Now let us prove the main problem. Let us prove it by negation. Let $\alpha \in CNF$, $\alpha = \bigwedge_{1 \le i \le n} C_i$ with $n \in \mathbb{N}^*$ and $\forall \sigma : \mathcal{P} \to \{0,1\}$ we have that $\sum_{k=1}^{n} \sigma(C_k) < \frac{n}{2}$.

Let n_T be the number of valuations in the α truth table. Thanks to property 4, we know that there are at least $n_T/2$ valuations that satisfy $C_k, \forall k\ [\![1,n]\!]$. This means that $\forall k\ [\![1,n]\!], \sum_{j=1}^{n_T} \sigma_j(C_k) \ge \frac{n_T}{2}$.

This implies that $\sum_{j=1}^{n_T} \sum_{k=1}^{n} \sigma_j(C_k) = \sum_{k=1}^{n} \sum_{j=1}^{n_T} \sigma_j(C_k) \ge \sum_{k=1}^{n} \frac{n_T}{2} = \frac{n * n_T}{2}$.

We have a contradiction since we also know that $\forall \sigma : \mathcal{P} \to \{0,1\}$ we have that $\sum_{k=1}^{n} \sigma(C_k) < \frac{n}{2}$ which implies that $\sum_{j=1}^{n_T} \sum_{k=1}^{n} \sigma_j(C_k) < \sum_{j=1}^{n_T} \frac{n}{2} = \frac{n * n_T}{2}$.

1.3) Prove that "For all formulas in CNF of n clauses, there exists a valuation where at least nr clauses are satisfied" does not hold anymore if $r > 1/2$.

Proof. Indeed the result is quite obvious since we have the proof we have done just before. The idea is that we can always find $\alpha \in CNF$ for which the result does not hold when $r > 1/2$.

An idea to construct such a α with $n = 2k$ clauses would be this way (with $p \in \mathcal{P}$): $\alpha = \bigwedge_{1 \le i \le n} C_i$ where $C_i = \begin{cases} p, \text{ if } i \text{ even} \\ (\neg p), \text{ if } i \text{ odd} \end{cases}$

Table of truth to get the idea for n even ($n = 4$):

p	C_1	C_2	C_3	C_4
0	1	0	1	0
1	0	1	0	1

If n is even, we can always construct formulas α in CNF where exactly $n/2$ valuations satisfy α so the result can not hold with $r > 1/2$ for all formulas in CNF.

For n odd we can show that this is not possible too. We can construct α the same way as before. Table of truth to get the idea for n odd :

p	C_1	C_2	C_3	C_4	C_5
0	1	0	1	0	1
1	0	1	0	1	0

Here we would like to find a $r = \frac{1}{2} + \epsilon$ independente of n such that $\frac{n+1}{2} \geq rn > \frac{n}{2}$. This means that we need to choose a ϵ such that $\forall n \in \mathbb{N}^*, \frac{1}{2n} \geq \epsilon > 0$ which is impossible. If we choose a tiny-tiny ϵ that satisfy the criteria until a certain big-big odd n, it will not hold for higher values odd of n. Thus for n odd too, we can always construct formulas $\alpha \in CNF$ where the result does not hold for $r > 1/2$.

□

Exercice 2

1.1) Prove that all natural number can be expressed in factorial representation. First, we need the show this property :

Property 5. $P_n : \sum_{k=0}^{n}(k * k!) + 1 = (n+1)!$

We can prove this quite easily using induction.

Proof. Case $n = 0 :\ 0 * 0! + 1 = (0+1)! = 1$

Let us suppose that P_n is true. We have to prove that P_{n+1} is true.

$$\begin{aligned}\sum_{k=0}^{n+1}(k * k!) + 1 &= (n+1) * (n+1)! + \sum_{k=0}^{n}(k * k!) + 1 \\ &= (n+1) * (n+1)! + (n+1)! \\ &= (n+2)!\end{aligned}$$

Thus the property is true $\forall n \geq 0$. □

Now we are going to prove the following property which is the one we want to prove here.

Property 6. P_n : *The natural number* n *can be written using factorial representation. This means that* $\exists k \in \mathbb{N}$ *and* $\forall i \in [\![0, k]\!]$, $\exists a_i \in [\![0, i]\!]$ *such that* $n = \sum_{i=0}^{k} a_i * i!$. *The factorial representation of* n *is represented by* $a_k ... a_1 a_0$ *where* k *is the smallest natural number that satisfies the previous result.*

Proof. Case $n = 0$: $0 = a_0 * 0! = \sum_{i=0}^{k} a_i * i!$ with $k = 0$ and $a_0 = 0$.

Let us suppose that P_n is true. We have to prove that P_{n+1} is true.

As this is our induction hypothesis, we know that $\exists k \in \mathbb{N}$ and $\forall i \in [\![0, k]\!]$, $\exists a_i \in [\![0, i]\!]$ such that $n = \sum_{i=0}^{k} a_i * i!$. For convenience, let us take the smallest k such that $\forall i \in [\![0, k]\!]$, $\exists a_i \in [\![0, i]\!]$ such that $n = \sum_{i=0}^{k} a_i * i!$. This way we do not work with useless zeros.

Let $i^* = min(i \in [\![0, k+1]\!] \mid a_i \neq i)$.

Here is a property we want to prove :

- If $i^* \leq k$, then $n + 1 = \sum_{i=0}^{k'} a'_i * i!$ where $k' = k$ and $a'_i = \begin{cases} 0, \forall i < i^* \\ a_i, \forall i > i^* \\ a_i + 1, \text{ if } i = i^* \end{cases}$
- If $i^* = k + 1$, then $n + 1 = 1 * (k + 1)! = \sum_{i=0}^{k'} a'_i * i!$ where $k' = k + 1$ and $a'_i = \begin{cases} 0, \forall i < i^* \\ 1, \text{ if } i = i^* \end{cases}$

Proving this previous property implies that $n + 1$ can be written in factorial representation.

Let us consider the first case where $i^* \leq k$. Then $\forall i \in [\![0, i^* - 1]\!]$, $a_i = i$ and $a_{i^*} \in [\![0, i^* - 1]\!]$.

$$\begin{aligned} n + 1 &= \sum_{i=0}^{k} a_i * i! + 1 \\ &= \sum_{i=i^*+1}^{k} a_i * i! + a_{i^*} * i^*! + \sum_{i=0}^{i^*-1} (i * i!) + 1 \\ &= \sum_{i=i^*+1}^{k} a_i * i! + a_{i^*} * i^*! + i^*! && \text{Because of the property 5} \\ &= \sum_{i=i^*+1}^{k} a_i * i! + (a_{i^*} + 1) * i^*! \\ &= \sum_{i=0}^{k'} a'_i * i! \end{aligned}$$

Where $k' = k$ and $a'_i = \begin{cases} 0, \ \forall i < i^* \\ a_i, \ \forall i > i^* \\ a_i + 1, \text{ if } i = i^* \end{cases}$

Let us now consider the second case where $i^* = k+1$, which is a special case. Here $\forall i \in [\![0, k]\!]$, $a_i = i$.

$$\begin{aligned} n + 1 &= \sum_{i=0}^{k} i * i! + 1 \\ &= (k+1)! \qquad \text{Because of the property 5} \\ &= \sum_{i=0}^{k'} a'_i * i! \end{aligned}$$

Where $k' = k + 1$ and $a'_i = \begin{cases} 0, \ \forall i < i^* \\ 1, \text{ if } i = i^* \end{cases}$

In both cases, $n + 1$ can be expressed in factorial representation.

To conclude, we have seen that the property is true when $n = 0$, we have proven that P_n implies that P_{n+1} is true. Thus the property stands $\forall n \geq 0$.

$\square$

1.2) Prove that this expression is unique (there is not two ways to express a natural number in factorial representation).

Proof. Let us suppose there exists n, a natural number such that $n = \sum_{i=0}^{k} a_i * i!$ and $n = \sum_{i=0}^{k'} b_i * i!$ where there is at least a i such that $a_i \neq b_i$.

Without loss of generality, we can write that $k' = k$ (because in the case of $k' > k$ we can say that all the $a_i = 0$, $\forall i \in [\![k+1, k']\!]$).

Let $l = max(i \in [\![0, k]\!] \mid a_i \neq b_i)$, $A = (a_0, a_1, \ldots, a_{l-1})$ and $B = (b_0, b_1, \ldots, b_{l-1})$. We define $\phi_{A,l}$ as follows : $\phi_{A,l} = \frac{1}{l!} \sum_{i=0}^{l-1} a_i * i!$

Hence we have that :

$$\begin{aligned}\sum_{i=0}^{k} a_i * i! = \sum_{i=0}^{k} b_i * i! \Rightarrow\ & a_l * l! + \sum_{i=0}^{l-1} a_i * i! = b_l * l! + \sum_{i=0}^{l-1} b_i * i! \\ \Rightarrow\ & a_l + (\frac{1}{l!}\sum_{i=0}^{l-1} a_i * i!) = b_l + (\frac{1}{l!}\sum_{i=0}^{l-1} b_i * i!) \\ \Rightarrow\ & \quad a_l + \phi_{A,l} = b_l + \phi_{B,l} \\ \Rightarrow\ & \quad (a_l - b_l) = (\phi_{B,l} - \phi_{A,l})\end{aligned}$$

By definition of a_l and b_l, we know that $(a_l - b_l) \in \mathbb{Z}\backslash\{0\}$. Moreover, by definition, we know that $\forall i \in [\![0, l-1]\!]$, $a_i \leq i$ and $b_i \leq i$. Thus :

Thus $0 \leq \phi_{A,l} \leq \frac{1}{l!}\sum_{i=0}^{l-1} i * i! \leq \frac{1}{l!}(\sum_{i=0}^{l-1}(i * i!) + 1 - 1) = \frac{l! - 1}{l!} = 1 - \frac{1}{l!} < 1$. Then $-1 < -\phi_{A,l} \leq 0$. We can deduce the same results for $\phi_{B,l}$, and especially $0 \leq \phi_{B,l} < 1$. Given those two results, we have that :

$$-1 < \phi_{B,l} - \phi_{A,l} < 1$$

This result is absurd since we have that $(a_l - b_l) = (\phi_{B,l} - \phi_{A,l})$ and that $(a_l - b_l) \in \mathbb{Z}\backslash\{0\}$.

Hence we showed that supposing there exists a natural number that can be expressed in two different ways in factorial representation implies a absurd result. Therefore the factorial representation of all natural numbers is different.

$\square$

1.3) Code description The idea is really simple, I just calculated the biggest number of the form $a_i * i!$ where $a_i \in [\![0, i]\!]$ such that the input number n minus $a_i * i!$ is positive or zero. From this point, I keep the value of i and a_i and I repeat the process with the remainder term $r = n - a_i * i!$ until $r = 0$. In the end, the factorial representation of the number is simply the a_i values with their respective positions i.

Exercice 3

1.1) Prove that all natural number can be expressed in Fibonacci representation.
Property 7. P_n : *The natural number n can be written using Fibonacci representation. This means that $\exists k \in \mathbb{N}$ and $\forall i \in [\![0, k]\!]$, $\exists a_i \in [\![0, 1]\!]$ such that $n = \sum_{i=0}^{k} a_i * f_{i+2}$ and $\forall i \in [\![0, k-1]\!]$, $a_i a_{i+1} = 0$. The Fibonacci representation of n is represented by $a_k...a_1a_0$ where k is the smallest natural number that satisfies the previous result.*

The very first thing we have to consider is that the Fibonacci number associated to a_i is f_{i+2}, not f_i. The other important thing, is the constraint $\forall i \in [\![0, k-1]\!]$, $a_i a_{i+1} = 0$. This implies that we can not have two consecutive a_i equal to one. Now that we have a better understanding of P_n, we are going to prove it.

Proof. Case $n = 0$: $0 = 0 * 1 = \sum_{i=0}^{k} a_i * f_{i+2}$ where $k = 0$ and $a_0 = 0$.

Let us suppose P_n is true. We want to prove that P_{n+1} is true. As this is our induction hypothesis, we know that $\exists k \in \mathbb{N}$ and $\forall i \in [\![0, k]\!]$, $\exists a_i \in [\![0, 1]\!]$ such that $n = \sum_{i=0}^{k} a_i * f_{i+2}$ and $\forall i \in [\![0, k-1]\!]$, $a_i a_{i+1} = 0$. For convenience, let us take the smallest k such that $\forall i \in [\![0, k]\!]$, $\exists a_i \in [\![0, 1]\!]$ such that $n = \sum_{i=0}^{k} a_i * f_{i+2}$ and $\forall i \in [\![0, k-1]\!]$, $a_i a_{i+1} = 0$. This way we do not work with useless zeros. We have $a_k = 1$ in all cases except if $n = 0$.

Let $i^* = min(i \in [\![0, k+1]\!] \mid a_i + a_{i+1} = 0)$. By this definition, we have that $a_{i^*} = 0$ and $a_{i^*+1} = 0$.

Here is a property we want to prove :

- If $i^* = 0$, then $n + 1 = \sum_{i=0}^{k'} a'_i * f_{i+2}$ where $k' = k = 0$ and $a'_0 = 1$.
- If $0 < i^* \leq k$, then $n + 1 = \sum_{i=0}^{k'} a'_i * f_{i+2}$ where $k' = k$ and $a'_i = \begin{cases} 0, \forall i < i^* \\ a_i, \forall i > i^* \\ 1, \text{if } i = i^* \end{cases}$
- If $i^* = k + 1$, then $n + 1 = \sum_{i=0}^{k'} a'_i * f_{i+2}$ where $k' = k + 1$ and $a'_i = \begin{cases} 0, \forall i < i^* \\ 1, \text{if } i = i^* \end{cases}$

Moreover the property states that in all cases $\forall i \in [\![0, k'-1]\!]$, $a'_i a'_{i+1} = 0$.

Proving the previous property implies that $n + 1$ can be written in Fibonacci representation. So let us prove it now :

The case $i^* = 0$ is trivial. This implies that $n = 0$, thus $n + 1 = 1 * 1 = 1 * f_2$. We obviously have $\forall i \in [\![0, k'-1]\!]$, $a'_i a'_{i+1} = 0$.

In the case $0 < i^* \leq k$, we can say that $\begin{cases} \forall i \in [\![0, i^*-1]\!], \ a_i a_{i+1} = 0 \\ \forall i \in [\![0, i^*-1]\!], \ a_i + a_{i+1} \neq 0 \\ a_{i^*-1} = 1 \end{cases}$ We can notice that i^* is necessarily different from k or $k - 1$.

In this case, n can be written in this way $n = \sum_{i=i^*+2}^{k} a_i * f_{i+2} + \sum_{i=0}^{i^*-1} a_i * f_{i+2}$. Here we can divide the problem in two parts : $i^* - 1$ is even and $i^* - 1$ is odd.

If $i^* - 1$ is even, $\forall i \in \mathbb{N}$ such that $2i \leq i^* - 1$, $a_{2i} = 1$ y $a_{2i-1} = 0$. Thus $n = \sum_{i=i^*+2}^{k} a_i * f_{i+2} + \sum_{i=0}^{(i^*-1)/2} f_{2i+2}$. As $f_1 = 1$, and by the property of the Fibonacci sequence $f_{n+2} = f_{n+1} + f_n$, we can show that :

$$
\begin{aligned}
n+1 &= \sum_{i=i^*+2}^{k} a_i * f_{i+2} + \sum_{i=0}^{(i^*-1)/2} f_{2i+2} + 1 \\
&= \sum_{i=i^*+2}^{k} a_i * f_{i+2} + \underbrace{\sum_{i=0}^{(i^*-1)/2} f_{2i+2} + f_1}_{=f_{i^*+2}} && \text{Property of the Fibonacci sequence} \\
&= \sum_{i=0}^{k'} a'_i * f_{i+2}
\end{aligned}
$$

If $i^* - 1$ is odd, $\forall i \in \mathbb{N}$ such that $2i + 1 \leq i^* - 1$, $a_{2i} = 0$ y $a_{2i+1} = 1$. Thus $n = \sum_{i=i^*+2}^{k} a_i * f_{i+2} + \sum_{i=0}^{(i^*-2)/2} f_{2i+3}$. As $f_2 = 1$, and by the property of the Fibonacci sequence $f_{n+2} = f_{n+1} + f_n$, we can show that :

$$
\begin{aligned}
n+1 &= \sum_{i=i^*+2}^{k} a_i * f_{i+2} + \sum_{i=0}^{(i^*-2)/2} f_{2i+3} + 1 \\
&= \sum_{i=i^*+2}^{k} a_i * f_{i+2} + \underbrace{\sum_{i=0}^{(i^*-2)/2} f_{2i+3} + f_2}_{=f_{i^*+2}} && \text{Property of the Fibonacci sequence} \\
&= \sum_{i=0}^{k'} a'_i * f_{i+2}
\end{aligned}
$$

In both cases ($i^* - 1$ even and odd), $k' = k$ and $a'_i = \begin{cases} 0, \forall i < i^* \\ a_i, \forall i > i^* \\ 1, \text{if } i = i^* \end{cases}$

In both cases $\forall i > i^*$, $a'_i = a_i$ so as we have that $\forall i \in [\![i^*+1, k-1]\!]$, $a_i a_{i+1} = 0$ we have that $\forall i \in [\![i^*+1, k'-1]\!]$, $a'_i a'_{i+1} = 0$. Moreover, $\forall i < i^*$, $a'_i = 0$ so $\forall i \in [\![0, i^*-1]\!]$, $a'_i a'_{i+1} = 0$. Finally, we know that $a'_{i^*} a'_{i^*+1} = 0$ because $a'_{i^*+1} = a_{i^*+1} = 0$ by definition of i^*. So $\forall i \in [\![0, k'-1]\!]$, $a'_i a'_{i+1} = 0$

If $i^* = k+1$, then if k is even $n+1 = \sum_{i=0}^{k/2} f_{2i+2} + 1 = \sum_{i=0}^{k/2} f_{2i+2} + f_1 = f_{k+3}$. If k is odd $n+1 = \sum_{i=0}^{(k-1)/2} f_{2i+3} + 1 = \sum_{i=0}^{(k-1)/2} f_{2i+3} + f_2 = f_{k+3}$.

So in both cases (k even and odd), $n+1 = \sum_{i=0}^{k'} f_{i+2}$ where $k' = k+1$ and $a'_i = \begin{cases} 0, \forall i < i^* \\ 1, \text{if } i = i^* \end{cases}$

Then obviously, as only $a'_{i^*} = 1$, $\forall i \in [\![0, k'-1]\!]$, $a'_i a'_{i+1} = 0$.

To conclude, we have seen that the property is true when $n = 0$, we have proven that P_n implies that P_{n+1} is true. Thus the property stands $\forall n \geq 0$.

□

1.2) Prove that this expression is unique (there is not two ways to express a natural number in Fibonacci representation). In order to proove this, we are going to use the same idea or method from the previous part.

Proof. Let us suppose there exists a natural number n such that $n = \sum_{i=0}^{k} a_i f_{i+2} = \sum_{i=0}^{k'} b_i f_{i+2}$ (like the previous exercice we can assume without loss of generality that $k' = k$) where $\forall i \in [\![0, k]\!]$, $a_i \in [\![0, 1]\!]$ and $b_i \in [\![0, 1]\!]$, where there exists at least a $i \in [\![0, k-1]\!]$ such that $a_i \neq b_i$. We have that $\forall i \in [\![0, k-1]\!]$, $a_i a_{i+1} = 0$ and $\forall i \in [\![0, k-1]\!]$, $b_i b_{i+1} = 0$.

Let $l = max(i \in [\![0, k]\!] \mid a_i \neq b_i)$. We know that $|a_l - b_l| = 1$. Let $A = (a_0, a_1, \ldots, a_{l-1})$ and $B = (b_0, b_1, \ldots, b_{l-1})$. We define $\beta_{A,l}$ as follows : $\beta_{A,l} = \frac{1}{f_{l+2}} \sum_{i=0}^{l-1} a_i f_{i+2}$.

$$\begin{aligned}
\sum_{i=0}^{k} a_i f_{i+2} = \sum_{i=0}^{k} b_i f_{i+2} \Rightarrow \quad & a_l f_{l+2} + \sum_{i=0}^{l-1} a_i f_{i+2} = b_l f_{l+2} + \sum_{i=0}^{l-1} b_i f_{i+2} \\
& \Rightarrow a_l + (\frac{1}{f_{l+2}} \sum_{i=0}^{l-1} a_i f_{i+2}) = b_l + (\frac{1}{f_{l+2}} \sum_{i=0}^{l-1} b_i f_{i+2}) \\
& \Rightarrow \qquad a_l + \beta_{A,l} = b_l + \beta_{B,l} \\
& \Rightarrow \qquad (a_l - b_l) = (\beta_{B,l} - \beta_{A,l})
\end{aligned}$$

Because of the constraint that $\forall i \in [\![0, k-1]\!]$, $a_i a_{i+1} = 0$ and because $f_1 = f_2 = 1$, if $l-1$ is even :

$$0 \leq \beta_{A,l} = \frac{1}{f_{l+2}}(\sum_{i=0}^{l-1} a_i f_{i+2}) \leq \frac{1}{f_{l+2}}(f_{l+1}+f_{l-1}+\cdots+f_4+f_2+f_1-1) = \frac{1}{f_{l+2}}(f_{l+2}-1) = 1-\frac{1}{f_{l+2}} < 1$$

If $l-1$ is odd, we obtain the same result :

$$0 \leq \beta_{A,l} = \frac{1}{f_{l+2}}(\sum_{i=0}^{l-1} a_i f_{i+2}) \leq \frac{1}{f_{l+2}}(f_{l+1}+f_{l-1}+\cdots+f_5+f_3+f_2-1) = \frac{1}{f_{l+2}}(f_{l+2}-1) = 1-\frac{1}{f_{l+2}} < 1$$

Then $-1 < -\beta_{A,l} \leq 0$. We can deduce the same results for $\beta_{B,l}$, and especially $0 \leq \beta_{B,l} < 1$. Given those two results, we have that :

$$-1 < \beta_{B,l} - \beta_{A,l} < 1$$

Finally, we have that $1 = |a_l - b_l| = \beta_{B,l} - \beta_{A,l} < 1$ which is absurd. Thus the Fibonacci representation is unique for all $n \in \mathbb{N}$.

$\square$

1.3) Code description The idea is really simple, I just calculated the biggest Fibonnaci number f_i such that the input number n minus f_i is positive or zero. From this point, I keep the value of i and f_i and I repeat the process with the remainder term $r = n - f_i$ until $r = 0$. In the end, the Fibonnaci representation of the number is simply the f_i values with their respective positions i. The construction of the code makes that, naturally, there will never be two consecutives 1 in the Fibonnaci representation.