



uOttawa

CSI 3520 - Concepts des langages de programmation

AUTOMNE 2022 SECTION A

PROFESSEUR KALONJI KALALA

---

## Devoir 4

---

Kien Do (ID: 300163370)

## 1. Réponse

Python

*# Puisque Python n'a pas introduit d'instructions à sélection multiple  
 ↳ (match/case) avant Python 3.10 en 2021, j'utiliserai des instructions  
 ↳ if/elif régulières pour cette partie du devoir.*

```
if k == 1 or k == 2:
    j = 2 * k - 1
elif k == 3 or k == 5:
    j = 3 * k + 1
elif k == 4:
    j = 4 * k - 1
elif k == 6 or k == 7 or k == 8:
    j = k - 2
```

Java

```
// Méthode "fall-through"
public class Main
{
    public static void main(String[] args) {
        int k = 1;
        int j = 0;

        switch(k) {
            case 1:
            case 2:
                j = 2 * k - 1;
                break;
            case 3:
            case 5:
                j = 3 * k + 1;
                break;
            case 4:
                j = 4 * k - 1;
                break;
            case 6:
            case 7:
            case 8:
                j = k - 2;
                break;
        }
    }
}
```

L'utilisation de plusieurs instructions à sélection multiple au lieu d'instructions conditionnelles if/else régulières augmente donc la fiabilité du programme.

## 2. Réponse

Segment de programme C sans **gotos** ni **breaks**.

```
#include <stdio.h>

int main()
{
    int j = -3;

    for (int i = 0; i < 3; i++)
    {
        if ((j + 2) == 3 || (j + 2) == 2)
        {
            j--;
        }
        else if ((j + 2) == 0)
        {
            j += 2;
        }
        else
        {
            j = 0;
        }

        if (j > 0)
        {
            j = 3 - i;
        }
    }

    return 0;
}
```


3. Affichez la pile avec toutes les instances d'enregistrement d'activation, y compris les chaînes statiques et dynamiques, lorsque l'exécution atteint la position 1 dans le programme squelette suivant. Supposons que bigsub est au niveau 1. (10pts)

```

function bigsub() {
  function a() {
    function b() {
      ... <-----1
    } // end of b
    function c() {
      ...
      b();
    } // end of c
    ...
    c();
  } // end of a
  ...
  a();
} // end of bigsub

```

Réponse:

ARI for B	Dynamic link
	Static link
	Return (to C)
ARI for C	Dynamic link
	Static link
	Return (to A)
ARI for A	Dynamic link
	Static link
	Return (to BIGSUB)
ARI for BIGSUB	Dynamic link
	Static link
	Return (...) 

4. Affichez la pile avec toutes les instances d'enregistrement d'activation, y compris les chaînes statiques et dynamiques, lorsque l'exécution atteint la position 1 dans le programme squelette suivant. Supposons que `bigsub` est au niveau 1. (10pts)

```

function bigsub() {
function a(flag) {
    function b() {
        ...
        a(false);
    } // end of b
    ...
    if (flag)
        b();
    else c();
    ...
} // end of a
function c() {
    function d() {
        ... <-----1
    } // end of d
    ...
    d();
    ...
} // end of c
    ...
    a(true);
    ...
} // end of bigsub

```

La séquence d'appel de ce programme à exécuter pour atteindre *d* est :

- *bigsub* appelle *a*.
- *a* appelle *b*.
- *b* appelle *a*.
- *a* appelle *c*.
- *c* appelle *d*.

Remarque: La réponse est sur la page suivante.

Réponse:

ARI for D	Dynamic link
	Static link
	Return (to C)
ARI for C	Dynamic link
	Static link
	Return (to A)
ARI for A	Parameter: flag=false
	Dynamic link
	Static link
	Return (to B)
ARI for B	Dynamic link
	Static link
	Return (to A)
ARI for A	Parameter: flag=true
	Dynamic link
	Static link
	Return (to BIGSUB)
ARI for BIGSUB	Dynamic link
	Static link
	Return (...)