

# Algorithmique et structures de données 2

## TP6

am@up8.edu

2022-2023

## 1 Consignes de rendu

TP à rendre pour le jeudi 20/04 12h (deadline fixe)

Un projet qui ne compile pas, qui termine sur une erreur de segmentation ou qui ne respecte pas les consignes ci-dessous ne sera pas corrigé.

- fichiers à rendre :
  - au minimum, fichiers `dijkstra.h` et `dijkstra.c` où vous définissez et implémentez les fonctions dont vous avez besoin pour calculer les distances entre un noeud et les autres noeuds du graphe selon l'algorithme de Dijkstra.
  - fichier `tp6.c` où vous testez les fonctions du module ci-dessus.
  - Makefile (cibles `all`, `clean`, `dist`)
- dépôt : archive `tp6_votre_nom.zip`
- vous pouvez ajouter un fichier `README.txt` au besoin (ne m'envoyez pas d'information complémentaire par mail / *via* mattermost)
- vous pouvez rendre le TP seule ou à deux. Dans ce cas, déposez la même archive avec vos deux noms sur vos deux moodles respectifs.

## 2 Réviser les tailles de variables en C

### 2.1 Fichier à examiner

Créez un fichier `test_variables.c` où vous déclarez, vous allouez la mémoire lorsque c'est nécessaires et initialisez des objets de type suivant :

- un `int` (ex : `int var_int = 3;`)
- un `float`
- un `char`
- une chaîne de caractères
- un `long int`
- une chaîne de caractères

- un pointeur sur un entier (ex : `int* var_p_int = malloc(sizeof(var_p_int));`)
- un pointeur de pointeur sur un entier
- un tableau de 5 entiers
- un objet de type `ma_struct` contenant deux entiers (appelés a et b) et un pointeur (appelé c) sur un objet de type `ma_struct`
- un pointeur sur un objet de type `ma_struct`

## 2.2 Vérifier les valeurs et les tailles

Pour chaque variable, renseignez dans un tableau :

- son nom, sa valeur, sa taille, la taille de son adresse.
- si c'est un pointeur : l'objet sur laquelle la variable pointe, la taille de la variable sur laquelle pointe l'objet, la valeur de l'objet sur lequel pointe la variable

Pour cela, compilez votre code avec `gcc -g` et utilisez `gdb` :

1. placez un breakpoint sur la dernière ligne de votre programme de façon à ce que toutes vos variables aient été déclarées.
2. lancez le programme (`run` (ou `r`))
3. pour chaque variable, utilisez `print` (ou `p`) pour afficher la variable `p var`, son adresse `p &var`, la taille de son adresse `p &var`, la taille de la variable `p sizeof(var)`, lorsque cela s'applique l'objet sur laquelle la variable pointe `p *var`, lorsque cela s'applique la valeur de l'objet sur lequel pointe la variable `p sizeof(*var)`.

## 2.3 Examiner la contiguïté

Vérifiez, toujours dans `gdb`, la contiguïté des données dans un tableau et dans un objet de type `struct`. Pour cela, affichez successivement les adresses de `tab`, `tab[1]`, etc. et de `var_struct->a`, `var_struct->b`, etc.

## 2.4 Vérifiez les réponses de l'exam 1

---

```

1 int main(){
2     int* p=NULL;
3     int tab[4];
4     printf("%ld, %ld, %ld\n",sizeof(tab), sizeof(&tab), sizeof(tab[0]))
5     ;
6     p = malloc(5*sizeof(int));
7     printf("%ld, %ld\n",sizeof(p), sizeof(*p));
8     p = malloc(4*sizeof(int));
9     printf("%ld, %ld\n",sizeof(p), sizeof(*p));
10    for (int i=0; i<5; i++){
11        if (i%2 == 0) { // i est pair
12            p[i]=i;
13        } else {
14            p[i] = 0;
15        }
16    }
17 }
```

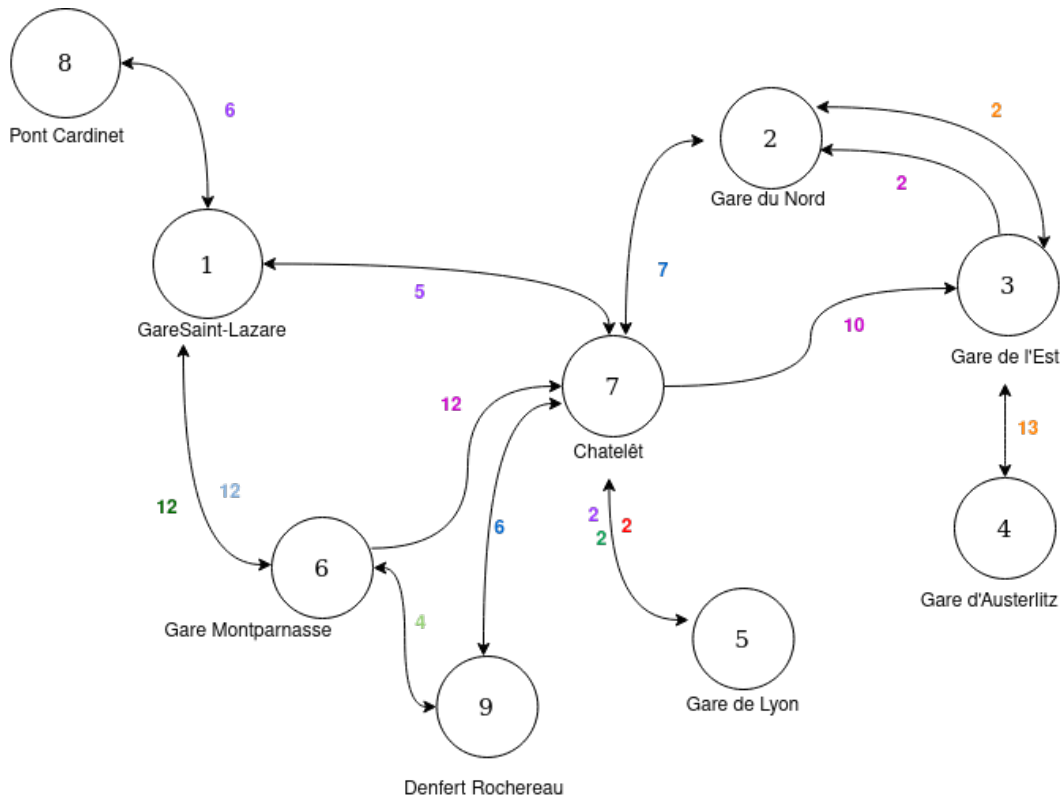
```
15     }
16     for (int i=0; i<5; i++){
17         printf("%d",p[i]);
18     }
19     free(p);
20     return 0;
```

---

Compilez et exécutez le code ci-dessus (voir le fichier exam1.c sur moodle).

1. vérifiez les réponses attendues pour les affichages. Si certains affichages vous surprennent, notez-les et demandez une explication.
2. modifiez le code pour ne faire afficher que les 4 premiers nombres impairs.

**SUITE PAGE SUIVANTE**



### 3 Dijkstra

#### 3.1 S'entraîner au calcul du plus court chemin

1. donner la matrice d'adjacence du graphe ci-dessus
2. donner le déroulé et le résultat de l'algorithme de Dijkstra depuis le noeud 2. Vérifiez votre résultat à l'aide du code du TP5.

#### 3.2 Finir le TP5

Vous devez finir le TP précédent (avec affichage des prédecesseurs) si cela n'a pas déjà été fait et le déposer dans la zone de dépôt correspondant au TP6 avant jeudi.

Faites un nouveau rendu complété et testant la configuration donnée dans la section ci-dessus.