

Nam Luân GUERIN

LSI1APP

TP5 C++ Les Fonctions

```
#include <iostream>
```

```
#include <cmath>
```

```
// 1) Fonction qui retourne le plus petit de deux entiers
```

```
int min(int a, int b) {
```

```
    if (a < b) {
```

```
        return a;
```

```
    }
```

```
    else {
```

```
        return b;
```

```
    };
```

```
}
```

```
// 1) Fonction qui retourne le plus grand de deux entiers
```

```
int max(int a, int b) {
```

```
    if (a > b) {
```

```
        return a;
```

```
    }
```

```
    else {
```

```
        return b;
```

```
    };
```

```
}
```

```
// 2) Fonction qui retourne le nombre nb a la puissance exp
```

```
int puissance(int nb, int exp) {
```

```
    int result = 1;    for(int i = 0; i <
```

```
    exp; i++) {        result *= nb;
```

```
    }
```

```
    return result;
```

```
}
```

```
// 3) Fonction qui verifie si le nombre est un entier bool
```

```
estEntier(const char* c) {
```

```
    bool estDecimal = false;
```

```
    int i = 0;
```

```

    if (c[0] == '-' || c[0] == '+') {
        i++;
    }

    for (; c[i] != '\0'; i++) {
        if (c[i] == '.') {
            estDecimal = true;
        } else if (c[i] < '0' || c[i] > '9') {
            return false;
        }
    }

    return true;
}

// 4) Fonction qui vérifie si une valeur est présente dans un tableau
int estPresenteTab(int val, int tab[], int lenTab){
    for (int i = 0; i < lenTab; i++) {
        if (tab[i] == val) {
            return i;
        }
    }
    return -1;
}

// 5) Compte le nombre d'occurrence dans une chaîne de caractère
int compterChar(char car, const char* chaine) {
    int compteur = 0;

    while (*chaine != '\0') {
        if (*chaine == car) {
            compteur++;
        }
        chaine++;
    }

    return compteur;
}

```

```

// 6) Affiche la table des valeurs de la fonction de l'enonce double
fonctionF(int x) {
    return sin(x) + log(x) - sqrt(x); // log(x) est le logarithme naturel
}

int main() {
//Exo 1
int a = 5;
    int b = 10;

    printf("Le min de %d et %d est %d\n", a,b,min(a, b));    printf("max
de %d et %d est %d\n", a,b,max(a, b));

    //Exo 2
int nb = 2;
    int exp = 3;
    printf("Puissance = %d\n", puissance(nb, exp));

    //Exo 3    char input[100];
printf("Entrez une valeur: ");
scanf("%s",input);    if
(estEntier(input)) {
    printf("L'entree %s est un nombre\n", input);
} else {
    printf("L'entree %s n'est pas un nombre\n", input);;
}

//Exo 4
int tab[] = {3, 4, 9, 12, 56};
int val = 0;

int longueur = sizeof(tab) / sizeof(tab[0]);
printf("Entrez la valeur recherchee");    scanf("%d",&val);
int position = estPresenteTab(val, tab, longueur); // Passe la taille calculée

if (position != -1) {
    printf("La valeur %d est presente a la position : %d\n", val, position); ;
} else {
    printf("La valeur %d n'est presente dans le tableau\n", val);
}

```

```

// Exo 5  char
chaine[100];  char
car;

printf("Entrez une chaine de caracteres : ");  scanf("%s",
chaine);

printf("Entrez le caractere a chercher : ");  scanf("
%c", &car);

int compte = compterChar(car, chaine);

printf("Le caractere '%c' apparait %d fois dans la chaine\n", car, compte);

//Exo 6
printf("x\tf(x)\n");

for (int x = 1; x <= 10; ++x) {
    printf("%d\t%.6f\n", x, fonctionF(x));
}
return 0;

}

```