

// Correction Exam Janvier 2019 Algo Mpi

#include<stdio.h>

#include<string.h>

// Exercice 1

//1 (3 pts)

```
int bien_parenthesee(char* s){
    int bp = 1, i=0, nb = 0;
    while (bp && i< strlen(s) ){
        if(s[i] == '('){
            nb++;
        } else if (s[i] == ')') {
            if(nb>0){
                nb--;
            } else {
                bp = 0;
            }
        }
        i++;
    }
    return (bp && !nb);
}
```

//2 (3 pts)

```
int nbre_facteurs(char* s){
    int fact = 0, i, nb = 1;
    for (i=1; i<strlen(s); i++){
        if(s[i] == '('){
            nb++;
        }
    }
}
```

```

        } else if (s[i] == ')') {
            nb--;
            if(nb==0){
                fact++;
            }
        }
    }
    printf("%d",fact);
    return fact ;
}

```

//Exercice 2 (4 pts)

```

procedure Fusion (E: tab1 tableau [1,N]: Entier,
Var i,j,k : Entier
Début
    i<-1
    j<-1
    k<-1

    tanque(i<=N et K<=M)
        si (tab1[i]<tab2[j]) alors
            tab3[k] = tab1[i]
            i<-i+1
        sinon si (tab1[i]>tab2[j]) alors
            tab3[k] = tab2[j]
            j<-j+1
        sinon

```

```

        tab3[k] = tab2[j]
        j<-j+1
        i<-i+1
    finsi
    k<-k+1
fintantque

si (i==n) alors
    pour j=j à M faire
        tab3[k] = tab2[j]
        k<-k+1
    finpour
sinon
    pour i=i à N faire
        tab3[k] = tab1[i]
        k<-k+1
    finpour
finsi
Finprocédure

```

// Exercice 3

// 1 (2.5 pts)

```

int nbJourBb (FILE* fp_bebe, int codeBb){
    int num,code; char s[100];
    rewind(fp);
    while(fgets(s,100,fp_bebe)){
        sscanf(s,"%5d",&code);
        if(code == codeBb){

```

```

        sscanf(s+25, "%8ld%8ld", &dn, &ds);
        if(!ds){
            printf("le bébé est encore dans");
            return -1 ;
        } else {
            return(getNbJours(dn, ds));
        }
    }
}

// 2.1 (2 pts)
void updateSortie(FILE* fp, long dateSortie, int codeBebe) {
    rewind(fp);
    while(fgets(s, 100, fp_bebe)){
        sscanf(s, "%5d", &code);
        if(code == codeBb){
            // on peut reculer de 12 ou 13 cases
            // S'il compte 2c pour \n ou 1
            fseek(fp, -12L, SEEK_CUR);
            fprintf(fp, "%8.8ld", dateSortie);
            break ;
        }
    }
}

// 2.2 (2.5 pts)
// On suppose que le bébé est sorti
int getPrixCouveuse(int codeBebe) {
    long de, ds;

```

```

int cj;
FILE* fp = fopen("couveuse.txt","r");
if(!fp){
    printf("Pbm ouverture ");
    exit(0);
} else {
    while(fgets(s,100,fp)){
        sscanf(s,"%5d",&code);
        if(code == codeBb){
            sscanf(s+5,"%8ld%8ld%5d",&de,&ds);
            return nbJourBb(de,ds)*cj;
        }
    }
}
return 0;
}

```

// 3 (3 pts)

```

int inTab(Medcin *t, int n,int code){
    int i;
    for (i=0;i<n;i++){
        if(t[i].code == code){
            return i;
        }
    }
    return -1;
}

```

```

int getStatsMedcins (FILE* fp, Medcin *t, long c

```

```

long date;
int nb=0,code,pos;
while(fgets(s,100,fp)){
    sscanf(s+5,"%5d%8ld",&code,&date);
    if(inInterval(dateDeb,dateFin,date))
        pos=inTab(t,n,code);
        if(pos == -1){
            t[n].nbVisite = 1;
            t[n].code = code;
        } else {
            t[pos].nbVisite ++;
        }
    }
}
return nb;
}

```

```

int main(){
    char s[20];

    gets(s);

    nbre_facteurs(s);
    // printf("%d",bien_parenthesee(s));
}

```