```
int main ()
{
int w;

if (w >= 50)
{
 printf("Congrats bro, you passed!\n");
 printf("We should throw a party for you");
}
else
{
 printf("You failed because you did not understand if and else");
 printf("\nSorry");


}
printf("Bye");
}
```

```
int main ()
{
int w;
w=60;
if (w >= 50)
{
 printf("Congrats bro, you passed!\n");
 printf("We should throw a party for you");
}
else
{
 printf("You failed because you did not understand if and else");
 printf("\nSorry");


}
printf("Bye");

}
```

```
int main ()
{
int w;
w=60;
if (w >= 50)
{
 printf("Congrats bro, you passed!\n");
 printf("We should throw a party for you");
}
else
{
 printf("You failed because you did not understand if and else");
 printf("\nSorry");


}

printf("Bye");

}
```

```
int main ()
{
int w;
w=40;
if (w >= 50)
{
 printf("Congrats bro, you passed!\n");
 printf("We should throw a party for you");
}
else
{
 printf("You failed because you did not understand if and else");
 printf("\nSorry");


}

printf("Bye");

}
```

```
int main ()
{
int w;
w= 40;
if (w >= 50)
{
 printf("Congrats bro, you passed!\n");
 printf("We should throw a party for you");
}
else
 {
   if(w<30)
   {
     printf("You failed because you did not understand if and else");
     printf("\nSorry");
   }
 }

printf("Bye");

}
```

```
int main ()
{
int w;
w= 40;
if (w >= 50)
{
 printf("Congrats bro, you passed!\n");
 printf("We should throw a party for you");
}
else
 {
   if(w<30) //Wrong
   {
     printf("You failed because you did not understand if and else");
     printf("\nSorry");
   }
 }

printf("Bye");

}
```

```
int main()
{
  int w=15;
if (w!=15)
    printf(“Hello”);
    printf(“\nHow are you?”);

}
```

```
int main()
{
  int w=15;
if (w!=15)
    {
    printf(“Hello”);
    }

printf(“\nHow are you?”);

}
```

No brackets? If you can’t see it doesn’t mean it doesn’t exist.
Le printf(“\nHow are you?”) va etre imprime independament du if

End of if-else

```
int main ()
{

int ans;

do{
    printf("I will ask for 3 volunteers later");
    printf("\nNow just enter a number");
    scanf("%d", &ans);
} while(ans>=0);


}
```

```
int main ()
{
int i;
i = 0;
while(i<=10){
    printf(“I’m repeating!”);
    printf(“%d”, i);
    i = i+1; // or i++

}
}
```

```
int main ()
{
int i;
for(i=0; i<=10; i++){
    printf(“I’m repeating!”);
    printf(“%d”, i);

}
}
```

```
int main ()
{
int i;
for(i=0; i<=10; i++){
    printf(“I’m repeating!”);
    printf(“%d”, i);

}
}
```

Or, you can use a break to break the loop, ie:

```
int i=-9;
while(1)
 {
  printf(“%d”, i);
  i=i+2;
  if (i==3)
     {
     break;
     }
  }
```

```
//Le tableau est le cmd.exe

int main()
{
int a, b;
a=2;
b = firstfunction(a);
printf(“%d %d”, a, b);
}

int firstfunction(int a)
{
int c;
a = a+1;
c = secondfunction(a);
printf(“%d %d\n”, a, c);
return c;
}


int secondfunction(int z)
{
z = z*z;
printf(“%d\n”, z);
return z;
}


}
```

```
//Le tableau est le cmd.exe

int main()
{
int a, b;
int *p;
p = &a;
a=2;
b = firstfunction(p);
printf("%d %d", a, b);
}

int firstfunction(int* pp)
{
      // now pp = p = &a, hence pp = &a; i.e. pp contient l'addresse
      //                  de a (alors pp 'pointe' a la variable a)
int c;
*pp = *pp+1;
c = secondfunction(pp);
printf("%d %d\n", *pp, c);
return c;
}


int secondfunction(int* ppp)
{
    //ppp = &a;
*ppp = *ppp*(*ppp);
printf("%d\n", *ppp);
return *ppp;
}
```

Faire le meme exercice avec return 2; au lieu de return *ppp; (derniere ligne).

```
c = b++;
IS EQUIVALENT TO:
```

```
c = b;
b++; // or b = b+1;
```

```
int main ()
{
```

```
int b = 2;
int c = (b++) + (b++) + (b++);
printf("%d %d", b, c);
}

Same as c = b+b+b;
b= b+1;
b= b+1;
b= b+1;
```

```
int main ()
{
```

```
    struct me
    {
        int ELG_grade;
        int MAT_grade;
        int GNG_grade;
    } typedef student;

    student Ziad;
    student Fadi;
    student* Ghazy = &Ziad;
    student* Malek = &Fadi;

    Ghazy->ELG_grade=99;
    Fadi.MAT_grade=100;

    if(Ziad.ELG_grade==Ghazy->ELG_grade)
        printf("Obvisouly it is true, this is always true.\n");
        else
        printf("Something's wrong\n");


    Malek->GNG_grade=100;

    Fadi.MAT_grade=99;

     printf("Fadi: GNG: %d, MAT:%d\n", Fadi.GNG_grade, Malek->MAT_grade);

     printf("I don't want Codeblocks to compile anymore, so:\n\n");

    Malek.ELG_grade=5;


}
```

1) Q: Pourquoi le programme ci-dessus ne va pas compiler?
Hint: Le probleme est dans: `Malek.ELG_grade=5;`
2) Q: Si nous enlevons cette ligne, que va imprimer le programme?

Reponses:
1) Car Malek est un pointeur. Pour acceder aux structures en utilisant des pointeurs, nous utilisons ->
2) Compliez et essayez :)