//41

ok

//42 Exercícios propostos 1

1. F
2. V
3. F
4. F
5. V
6. F
7. F
8. F
9. V
10. F
11. V
12. F
13. F
14. F
15. V
16. V
17. V
18. V

//43 Exercícios propostos 2

2.1 Ponto e vírgula após cabeçalho da função

2.2 Função void não pode haver return.

2.3 Protótipo é diferente cabeçalho da função.

2.4 Protótipo é diferente cabeçalho, um é void outro é int.

2.5 Função sem nome.

2.6 Falta tipo da variável y.

//44 Exercícios propostos 3

1. Nada.
2. Já passei em C

Olá

1. Nada.

//45

int abs(int x) {  
if(x<0)  
return x\*(-1);  
else  
return x;  
}

//46

#include<stdio.h>  
int pot(int b, int e){  
 int i, r=1;  
 for(i=0;i<e;i++){  
 r\*=b;  
 }  
 return r;  
}  
float VAL(float x, int n, float t){  
 int i;  
 for(i=0;i<n;i++){  
 VAL+=(float)x/pot(1+t, i);  
 }

return VAL;  
}

//47

long int n\_segundos(int n\_horas){

long int seg;

seg=n\_horas\*3600:

return seg;

}

//48  
  
long int tempo(int n\_horas; char o){  
 long int tempo;  
 if(o=='s')  
 tempo=n\_horas\*3600:  
 else if(o=='m')  
 tempo=n\_horas\*60;  
 else if(o=='h')  
 tempo=n\_horas;  
 return tempo;  
}  
  
//48  
  
long int tempo(int n\_horas; char o){  
 long int tempo;  
 switch(o){  
 case 's':  
 tempo=n\_horas\*3600:  
 break;  
 case 'm':  
 tempo=n\_horas\*60;  
 break;  
 case 'h':  
 tempo=n\_horas;  
 break;  
 }  
 return tempo;  
}  
  
long int tempo(int n\_horas; char o){  
 long int tempo;  
 switch(o){  
 case 's':  
 tempo=n\_horas\*3600:  
 return tempo;  
 case 'm':  
 tempo=n\_horas\*60;  
 return tempo;  
 case 'h':  
 tempo=n\_horas;  
 return tempo;  
 }  
}  
  
//49

float Max(float x, float y, float z){  
 if(x>y&&x>z)  
 return x;  
 if(y>x&&y>z)  
 return y;  
 if(z>x&&z>y)  
 return z;  
}

//50

int impar(int x){  
 if(x%2!=0)  
 return 1;  
 else  
 return 0;  
}

//51

int entre(int x; int lim\_inf; lim\_sup){  
 if(x>=lim\_inf&&x<=lim\_sup)  
 return 1;  
 else  
 return 0;  
}

//52

11.   
  
11.1   
  
int isdigit(char c) {  
  
return (c>='0' && c<='9');   
}  
11.2   
  
int isalpha(char c) {  
  
return (c>='a' && c<='z' || c>='A' && c<='z');

}   
  
11.3   
  
int isalnum(char c) {  
  
return isalpha(c) || isdigit(c);

}   
  
11.4   
  
int islower(char c) {  
  
return (c>='a' && c<='z');

}  
  
11.5   
  
int isupper (char c){   
  
return (c>='A' && c<='z');

}  
  
11.6   
  
int isspace(char c) {  
  
return c==' ' || c=='\t';

}  
  
11.7   
  
char tolower(char c) {

if isupper(c)

return c + 'a' -'A';

else return c;   
}  
11.8   
  
char toupper(char c) {

if islower(c)

return c + 'A' -'a';

else

return c;   
}

//53   
  
int is\_square(int x, int y) {  
  
return x==y\*y;

}

//54

int Minus(int valor) {

if (valor<0)

return valor;

else return -valor;   
}

//55

int is\_special(int x) {  
  
return 2\*x:-x\*x;   
}  
//56

int Cubo(int x) (}  
  
return x\*x\*x;   
  
}  
  
//57

int IsVogal(char ch) {  
 if(ch=='a'||ch=='A'||ch=='e'||ch=='E'||ch=='i'||ch=='I'||ch=='o'||ch=='O'||ch=='u'||ch=='U'||)  
 return 1;  
 else  
 return 0;  
}