

Especificação da linguagem:

- Variáveis:

- "int" – Número inteiro, ex: "int numero = 1;"
- "float" – Número real, ex: "float numero = 1.5;"
- "char" – Caractere único, ex: "char caracter = 'a';"
- "bool" – Valor booleano, ex: "bool valor = TRUE;"
- "string" – String de caracteres, ex: "string frase = "Boa tarde";"
- "[]" – Array de variáveis, ex: "int lista[5] = {1,2,3,4,5};"

- Constantes literais:

- "const" -ex: const int = 5;

- Operadores:

- "+" – soma, usado pelas variáveis "int", "float" para somar números e na variável "string" e "char" para concatenar
- "-" – subtração, usada pelas variáveis "int", "float" para subtrair números
- "*" – multiplicação, usada pelas variáveis "int", "float" para multiplicar números
- "/" – divisão, usada pelas variáveis "int", "float" para dividir números
- "%" – resto, usada pelas variáveis "int", "float" para achar o resto da divisão
- "++" – adiciona 1 ao valor da variável "int" ou "float"
- "--" – subtrai 1 do valor da variável "int" ou "float"
- "=" – igualar, usada por todas variáveis para atribuir um valor
- "==" – checar igualdade, usada para checar se um elemento é igual ao outro, usada por todas variáveis
- "!=" – checar desigualdade, usada para checar se um elemento é diferente do outro, usada por todas variáveis
- "<" – checar se um elemento é menor que outro, usada por "int" e "float"

- "<=" – checar se um elemento é menor ou igual que outro, usada por "int" e "float"
- ">" – checar se um elemento é maior que outro, usada por "int" e "float"
- ">=" – checar se um elemento é maior ou igual que outro, usada por "int" e "float"
- "&" – operador lógico de conjunção, usada por 2 variáveis "bool"
- "|" – operador lógico de disjunção, usada por 2 variáveis "bool"
- "!(bool)" – operador de negação de valor "bool"

- Delimitadores:

- "()" – usado nas keywords, na chamada e definição de funções e em operações matemáticas
- "{}" – usado em definir espaço das funções e das keywords e para atribuir valores a um array
- "[]" – usado para manusear arrays
- ";" – usado no fim de cada linha exceto em uso de keywords e definição de funções
- " " – usado para definir strings ou chars
- ", " – usado para separar certos componentes

- Keywords:

- "for" – keyword de contagem até certa condição, ex :

```
for( int i=0, i<aux, i++);
{
};
```
- "while" – keyword de repetição até certa condição, ex:

```
while ( i < 3);
{
};
```

- “if” – keyword de checagem booleana, ex:

```
“if (i<3);  
{;  
};”
```

-“print” – mostra na tela algo, ex:

```
“print(“oi”);”
```

-“scan” – escaneia input do usuario, ex:

```
“scan(“%d”, a);”
```

-Observações:

- Deve ser adicionado um “;” ao final de toda linha do programa.

- Não pode declarar várias variáveis em uma única chamada, ex:

```
Int a,b,c=0;
```

-Exemplos de código:

```
1)  int data[];  
    int interval;  
    int i;  
    int n;  
    scan("%d", n);  
    for (i = 0, i<n, i++);  
    {;  
        scan("%d", data[i]);  
    };  
    for (interval = n / 2, interval > 0, interval = interval/2);  
    {;  
        for (i = interval, i < n, i++);  
    };
```

```

int temp = data[i];

int j;

for (j = i, j >= interval & data[j - interval] > temp, j = j - interval);

{
    data[j] = data[j - interval];

};

data[j] = temp;

};

};

for (i = 0, i < size, i++) ;

{
    printf("%d ", data[i]);

};

```

2)

```

int entrada;

scanf("%d", entrada);

int i;

int proxtermo;

int t1 = 0;

int t2 = 1;

for(i=0,i<entrada,i++);

{

    print("%d", t1);

    proxtermo = t1 + t2;

    t1 = t2;

    t2 = proxtermo;

};

```

