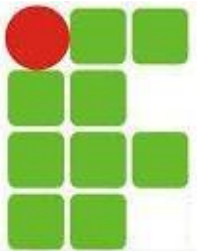
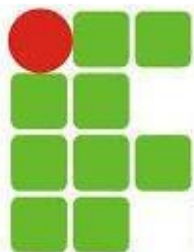


ALGORITMOS

Professor: Diego Oliveira

Aula 12 -
Laços de Repetição (WHILE)



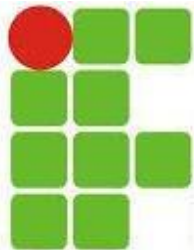


WHILE

- O laço de repetição WHILE, em português ENQUANTO, é executado enquanto uma condição for verdadeira:

```
11 int main()
12 {
13
14     int i = 1;
15
16     while(i<=10){
17         printf("i=%d \n", i++);
18     }
19
20     return 0;
21 }
```

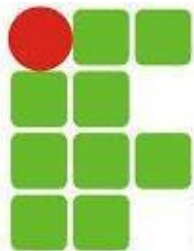
```
i=1
i=2
i=3
i=4
i=5
i=6
i=7
i=8
i=9
i=10
```



WHILE

- Um erro muito comum quando se utiliza o WHILE é não atualizar a variável que está sendo verificada, o que causa um laço infinito e o programa ‘trava’.
- Por isso o ‘i’ no código anterior está sendo incrementado de um em um DENTRO DO LAÇO!



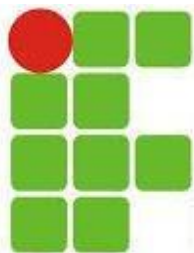


WHILE

- O WHILE também pode ser utilizado para executar laços dos quais não sabemos o número de repetições necessárias, como por exemplo um MENU:

```
12  int main()
13  {
14      int opcao = 0;
15
16      while(opcao!=3){
17          printf("1-START GAME \n");
18          printf("2-LOAD GAME \n");
19          printf("3-EXIT GAME \n");
20          scanf("%d",&opcao);
21      }
22
23      return 0;
24  }
```



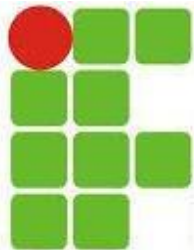


WHILE

- Saída do slide anterior para entradas '1', '2' e '3':

```
1-START GAME
2-LOAD GAME
3-EXIT GAME
1
1-START GAME
2-LOAD GAME
3-EXIT GAME
2
1-START GAME
2-LOAD GAME
3-EXIT GAME
3
```



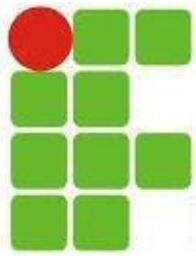


WHILE

- Um outro exemplo do uso do WHILE seria para calcular a média de uma turma, da qual não sabemos ainda o número de alunos:

```
12 int main()
13 {
14     int numeroAlunos = 0;
15     double mediaTurma = 0;
16     double mediaAluno = 0;
17
18     while(mediaAluno != -1){
19         printf("Digite a média do aluno: \n");
20         scanf("%lf",&mediaAluno);
21         if(mediaAluno != -1){
22             numeroAlunos++;
23             mediaTurma += mediaAluno;
24         }
25     }
26
27     mediaTurma = mediaTurma / numeroAlunos;
28     printf("A média da turma é: %lf", mediaTurma);
29
30     return 0;
31 }
```





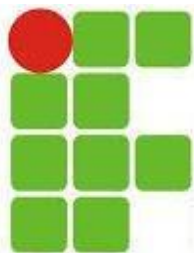
WHILE

- Saída do código anterior:

```
▼ ↗ 📄 ⚙️ 🖨️
Digite a média do aluno:
60
Digite a média do aluno:
80
Digite a média do aluno:
-1
A média da turma é: 70.000000
```



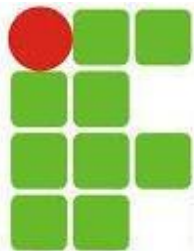
- Observe que foi definido para sair do laço caso a média digitada fosse -1, pois não é uma nota válida



Números Aleatórios

- Em Jogos se utiliza muito a geração aleatória de números para criar objetos em posições aleatórias no cenário ou, mais recentemente, gerar cartas em LOOT
- Em C podemos utilizar o seguinte código:

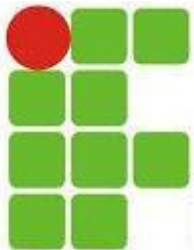
```
9  #include <stdio.h>
10 #include <stdlib.h>
11 #include <time.h>
12
13 int main()
14 {
15     srand(time(NULL));
16
17     int aleatorio = (rand() % 100) + 1;
18     printf("Número aleatório entre 1 e 100: %d\n", aleatorio);
19
20     return 0;
21 }
```



Exercício

- Faça um jogo no qual será gerado um número aleatório entre 1 e 10000.
- O usuário poderá chutar até 20 vezes
- Caso ele chute um número menor que o gerado aleatoriamente, imprima “chute um número MAIOR”, se ele digitar um maior imprima “chute um número MENOR”.
- Quando ele acertar, uma mensagem de PARABÉNS deve ser impressa
- Caso ele não acerte em 20 tentativas imprima “Infelizmente você perdeu!”





Perguntas?

