

SME0230 - Introdução à Programação de Computadores

Professora: Marina Andretta (andretta at icmc.usp.br)

Estagiário PAE: Pâmela Cândida (pamela at icmc.usp.br)

Monitor: Matheus Doretto Compri (matheusdc at grad.icmc.usp.br)

Forma de entrega: O exercício desta aula prática deverá ser entregue por e-mail para

- andretta at icmc.usp.br
- pamela at icmc.usp.br
- matheusdc at grad.icmc.usp.br

Ele deverá estar no formato .c e o nome do arquivo deverá ser numeroUsp.c. Exemplo: 6711776.c

O título da mensagem será: IPC2011_aula11

Observações importantes:

1. não serão recebidos trabalhos após dia 30/05 23h59;
2. o aluno que enviar o trabalho fora do padrão ou que não enviar para os e-mails especificados receberá zero como nota do referente exercício da aula prática.

Aula Prática 11 - Maratona de Programação - Usando funções

Nesta aula iremos correr atrás de ficarmos bons em programação. Programar não é só escrever um código que resolve um problema, mas codificar a solução para qualquer instância desse problema e sermos fiéis ao formato de saída esperada. Vamos treinar isso usando um problema conhecido por quem gosta de maratona de programação. Entre no site

<http://uva.onlinejudge.org/>

e então

Browse Problems -> Contest Volumes -> Volume C -> 10018 - Reverse and Add

e veja o problema 10018 - Reverse and Add. Implemente a solução para este problema seguindo rigidamente o enunciado, entrada e saída esperados. Na sua implementação devem constar as duas seguintes funções:

```
int is_palindrome(long long number);
```

```
long long reverse_and_add(long long number);
```

Para resolver esse problema, talvez você precise transformar um número em uma cadeia de caracteres (“lld” designa o formato de número “long long”):

```
sprintf(str, "%lld", number);
```

ou transformar uma cadeia de caracteres em um número:

```
sscanf(str, "%lld", &number);
```

Terminado seu problema, crie uma série de casos de teste para validá-lo. Para facilitar essa validação, lembre-se que na primeira aula de laboratório vimos redirecionamento de entrada e saída. Faça seu cadastro no site do UVA e quando achar que seu programa funciona, clique em “Submit” e envie seu .c. No menu “My Submissions”, você verá se seu programa realmente resolve todas as instâncias de teste do UVA “== Accepted” (são muitas!). Boa sorte!