

# Trabalho Prático 13

## Praia

### Algoritmos e Estruturas de Dados I

Prazo: **16 de novembro** de 2017

## 1 Introdução

A melhor época do ano para Joãozinho e Zezinho são as férias. Durante esse período, a família toda viaja para a casa no litoral e todos os dias eles vão para a praia.

Eles frequentam sempre a mesma praia e, como nas férias muitas outras famílias também resolvem viajar para o litoral, muitas vagas já estão tomadas, então, cada dia, Pedro estaciona o carro em um local diferente ao longo da praia e instala sua família em um ponto em frente ao carro.

Joãozinho é muito comilão, e adora de tomar sorvete na praia. Contudo, alguns dias acontece de nenhum sorveteiro passar pelo local onde eles estão. Intrigado com isto, e não querendo mais ficar sem tomar seu sorvete semanal, Joãozinho, em um dia de sorte, perguntou a um sorveteiro o que acontecia. Acontece que cada sorveteiro passa o dia percorrendo uma mesma região da praia, indo e voltando.

Não satisfeito, Joãozinho utilizou a ferramenta Poodle e rapidamente encontrou tanto a informação de início e fim das regiões percorridas por cada um dos sorveteiros quanto um sistema que mostrava em tempo real os locais com vagas ocupadas na frente da praia.

Com base nestes dados, Joãozinho quer descobrir os locais da praia onde o pai dele deve parar o carro, de forma que haja uma vaga e pelo menos um sorveteiro passando naquele local. Só que o volume de dados é muito grande, e Joãozinho está pensando se seria possível utilizar o computador para ajudá-lo nesta tarefa. Por isso, ele pediu a sua ajuda.

## 2 Tarefa

Você deve escrever um programa que leia tantos os dados obtidos pelo Joãozinho quanto as regiões que já estão ocupadas e imprima uma lista de intervalos da praia por onde passa pelo menos um sorveteiro e exista ao menos uma vaga.

### 3 Entrada

Seu programa deve ler vários conjuntos de teste. A primeira linha de um conjunto de teste contém três inteiros não negativos,  $P$ ,  $S$  e  $V$ , que indicam, respectivamente, o comprimento em metros da praia ( $0 < P \leq 100000$ ), o número de sorveteiros e o número de regiões com vagas ocupadas. Seguem-se  $S$  linhas, cada uma contendo dois números inteiros  $U_v$  e  $W_v$  que descrevem o intervalo de trabalho de cada um dos sorveteiros, em metros contados a partir do início da praia. As próximas  $0$  linhas são compostas, também, de dois números inteiros  $U_v$  e  $W_v$ , que descrevem o intervalo de **vagas ocupadas**, em metros contrados a partir do início da praia. O final da entrada é indicado por  $P = S = V = 0$ . **Para facilitar, as posições da praia começam em 0 e terminam em  $P - 1$ .**

### 4 Saída

Para cada conjunto de teste da entrada seu programa deve produzir uma lista dos intervalos da praia que possuem **vagas livres** e são servidos por pelo menos um sorveteiro. Cada intervalo da lista deve aparecer em uma linha separada, sendo descrito por dois números inteiros  $U_i$  e  $V_i$ , representando respectivamente o início e o final do intervalo. A ordem dos intervalos deverá ser do mais próximo do início da praia ( $U_i$  menor) até o mais próximo do fim da praia ( $U_i$  maior). Quando não houver intervalo livre, seu programa deverá imprimir "Nao ha vagas", sem acentos e sem aspas. Na dúvida, siga o exemplo abaixo.

### 5 Exemplo

#### Entrada

```
500 2 1
0 250
300 310
100 200
1000 1 1
0 100
0 200
0 0 0
```

#### Saída

```
0 99
201 250
300 310
Nao ha vagas
```

Outros exemplos estão disponíveis na página deste trabalho no Moodle.

## 6 Avisos

Avisos mandatórios para o envio do trabalho:

- O código deve ser escrito em linguagem C
- As entradas que serão utilizadas para teste não conterão erros, então não será necessário testar a validade das mesmas
- Não utilize chamadas da função `system` (por exemplo, `system("pause")`) pois essas podem variar de acordo com o sistema operacional e os programas instalados da máquina onde o programa está rodando
- Deixe seu código bem comentado para facilitar a correção
- Não utilize espaços ou caracteres especiais nos nomes dos arquivos. Utilize apenas os caracteres de A a Z (tanto maiúsculas como minúsculas) sem acento, os números de 0 a 9 e os caracteres - (hífen), \_ (*underscore*) e . (ponto final)
- Utilize a extensão `.c` para arquivos de código e `.h` para arquivos de cabeçalho, quando aplicar
- Se for submeter os arquivos via upload, não envie um arquivo comprimido (por exemplo, `.zip`, `.rar`, etc.). Utilize os diversos campos da aba “Submissão”, um para cada arquivo
- Envie apenas os arquivos `.c` e `.h`
- Não copie o trabalho de algum colega ou baixe da internet. Lembre-se que o prejudicado será você pois o aprendizado obtido nessa disciplina será utilizado durante diversas outras etapas do seu curso

## 7 Dicas

Dicas importantes para o desenvolvimento deste trabalho:

- Utilize a função `scanf` para ler as entradas do usuário
- Utilize a função `printf` para imprimir os resultados das operações
- Utilize um arranjo para guardar as posições da praia
- Muito cuidado com o início e fim dos intervalos, que devem ser respeitados
- Lembre-se que a praia começa na posição 0 e termina em  $P - 1$
- Qualquer dúvida que tiver, utilize o fórum de dúvidas no Moodle. Inicie o assunto do tópico com a tag [TP13]
- Caso prefira, participe das monitorias toda quarta das 17h às 18h na sala 2012

## 8 Checklist

Checklist não exaustiva de passos até a entrega do trabalho:

1. Pesquise o funcionamento das funções citadas acima para facilitar o uso
2. Implemente e compile o programa
3. Teste para o exemplo dado acima. Compare as saídas para garantir o funcionamento correto
4. Faça o mesmo do item anterior para os exemplos disponibilizados na página deste trabalho no Moodle
5. Envie o trabalho pelo Moodle, onde ele será testado automaticamente para todos os casos disponíveis
6. Caso algum teste dê errado, volte ao passo 2

Bom trabalho e divirta-se!