



90 minutos

Nome

n.º

Regras

Qualquer tentativa de fraude conduzirá à anulação da prova para todos os intervenientes.

Consulta: Apenas em papel. Não são permitidos meios electrónicos (calculadora, telemóvel, ...).
As respostas devem ser obrigatoriamente efectuadas na folha fornecida.

1 Questão (75%)

Implemente um simulador de uma estação de notícias com várias equipas de reportagem espalhadas pelo mundo. Cada uma das equipas envia periodicamente as notícias para a central. Na central, existe um editor que revê as notícias por ordem de chegada e decide quais devem ser publicadas.

Na sua solução, represente cada uma das equipas de reportagem como um processo (devem existir 10 equipas). Cada equipa deverá comunicar com o processo principal através de um *pipe* (i.e. haverá tantos *pipes* como equipas de reportagem). Cada equipa deve enviar uma notícia a cada 1-3 segundos.

O processo principal será o editor. Deverá ficar encarregue de monitorar todos os *pipes* em simultâneo, ler as notícias recebidas e aceitar apenas 10%, imprimindo-as para a consola.

Após receber Ctrl+C o programa deverá terminar eliminando todos os recursos. Esta opção só estará disponível após todos os processos criados. O processo principal apenas deverá terminar após todos os processos filho terminarem.

Exemplo do output:

```
NewsTeam 0 Submitting news: "home team wins"
NewsTeam 5 Submitting news: "food for the soul"
NEWS FLASH: team [1]: home team wins
NewsTeam 4 Submitting news: "an amazing discovery"

^C pressed. Shutting down news team 2...
^C pressed. Shutting down news team 4...
^C pressed. Editor shutting down...
```

Complete o código que se segue escrevendo nos locais a cinzento.

```
(...)  
#define N_NEWSTEAMS 10  
#define EXAMPLE_NEWS 5  
#define NEWS_MAX_SIZE 512  
  
int main_pid;  
int pipes [N_NEWSTEAMS][2];  
  
// Just some news used as an example  
char example_news[EXAMPLE_NEWS][NEWS_MAX_SIZE] = {  
    "all about the money",  
    "food for the soul",  
    "home team wins",  
    "an amazing discovery",  
    "newest tec came out"  
};
```

```

// News structure that will be used to send news through the pipes
typedef struct _news {
    int team;
    char newsText[NEWS_MAX_SIZE];
} news;

news news_data;

// Treatment of signal generated by Ctrl+C
// Editor waits for all the NEWSTEAMS to finish and removes resources (closes all open
// pipes); NEWSTEAMS clean used resources (close all open pipes). Each process prints a
// message to the screen (see example output)

void sigint(int signum) {
    // *** COMPLETE ***

    // *** END COMPLETE ***
    exit(0);
}

// Code run by NEWSTEAMS
void newsTeam(int channel[], int i){
    news_data.team = i;
    srand(i);
    // *** COMPLETE: Handle CTRL-C ***

    // *** END COMPLETE ***

    while(1){
        //pick a random news content
        strcpy(news_data.newsText , example_news[rand() % EXAMPLE_NEWS]);
        printf("News team %d Submitting news: \"%s\" \n", news_data.team,
news_data.newsText);
        // *** COMPLETE: Send news to pipe and sleep between 1 to 3 seconds ***

        // *** END COMPLETE ***
    }
}

```

Nome:_____

Número:_____

```
// Main
int main() {
    main_pid = getpid();
    // *** COMPLETE: Ignore CTRL-C ***

    // *** END COMPLETE ***

    // *** COMPLETE: Create NEWSTEAMS (processes and pipes) ***

    // *** END COMPLETE ***

    // Editor
    fd_set read_set;
    int r;
    // *** COMPLETE: Handle CTRL-C ***

    // *** END COMPLETE ***

    while (1) {
        // *** COMPLETE: Read from Pipes, publish only 10% ***
```

```
    // *** END COMPLETE ***  
} // While(1)  
return 0;  
}
```

Nome:_____

Número:_____

2 Questão (15%)

Durante as aulas práticas foram usados 3 conjuntos de funções diferentes para trabalhar com semáforos, declaradas em diferentes *header files* – *sem.h*, *semaphore.h* e *semLib.h*. Qual a diferença entre eles e em que situações cada um deles se deve usar?

3 Questão (10%)

No projeto prático, a utilização de um *named pipe* permite ao *Configuration Console* enviar comandos para alteração de configuração para o *HTTP Server*. Diga o que seria necessário mudar no sistema para que fosse possível a *Configuration Console* receber uma notificação do resultado dos comandos.