



DEPARTAMENTO DE FORMAÇÃO DE
ENGENHEIROS NAVAIS - RAMO DE ARMAS E
ELECTRÓNICA

3105 – SISTEMAS DIGITAIS II

4º ANO AEL

1ª Repetição Escrita de 2004/2005

Leia atentamente o enunciado. Seja breve nas respostas, mas justifique-as convenientemente. Por favor, use letra legível! Quando apresentar programas comente-os devidamente. Bom trabalho neste exame

I

Como decerto saberá, alguns dos seus camaradas estão a trabalhar num UAV (Unmanned Aerial Vehicle) que está a desenvolvido aqui na Escola Naval. Vamos ao longo desta repetição escrita desenvolver algum do software necessário para esse projecto (software esse que até é bastante parecido com aquele que já desenvolveu para o PCROBOTS).

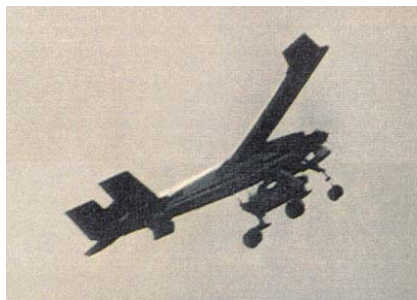
Uma das funções mais básicas consiste em ser capaz de percorrer um conjunto de “waypoints” previamente definidos. Para isso, o UAV dispõe de um GPS que indica a posição em cada ponto, uma agulha de fluxo magnético que indica a direcção em que o UAV está apontado, e um leme de direcção que faz com que o UAV vire. Para simplificar, vamos usar como coordenadas geográficas coordenadas de grid cartesiana x,y, em “unidades de grid” inteiras.

Para tornar o nosso sistema muito flexível, vamos guardar os waypoints que queremos percorrer numa lista ligada, usando apontadores. Cada elemento dessa lista terá a seguinte estrutura:

```
typedef struct x{  
    long    x,y;        /* Coordenadas do waypoint */  
    int     tolerance; /* Erro máximo admissível para a aproximação a (x,y) */  
    struct x *next;    /* apontador para a próxima ficha */  
}waypoint;
```

A interface com o GPS, agulha magnética e leme é feita chamando as rotinas que têm os seguintes protótipos:

```
void Get_Position(long &x, long &y); /* x,y são as coordenadas de grid actuais*/  
void Get_Direction(int &dir);      /* dir é a direcção em que o UAV está apontado, em  
                                     graus) */  
void Set_rudder(int rud);          /* (rud=0 => leme a meio), (rud=-1 => BB leme), e  
                                     (rud=1 => EB leme) */
```



I.a) Só para começar, escreva (em C) um programa auxiliar, para correr num PC vulgar, que converta de coordenadas de latitude e longitude para coordenadas de grid. Assuma que o centro da grid está em 39° 0' 0'' N, 8° 0' 0'' W, que cada segundo de latitude corresponde a 10 unidades de grid, e que cada segundo de longitude corresponde a 7 unidades de grid.

I.b) Escreva uma rotina em C que dada uma coordenada (xref,yref), calcule qual a direcção que o UAV deverá seguir para chegar lá. A rotina deverá chamar-se “**Calc_direction**”, deverá receber dois parâmetros de entrada do tipo “long” (xref e yref), e deverá devolver um valor de retorno inteiro que é a direcção a tomar.

I.c) Escreva uma rotina em C que dada uma direcção alvo, aponte o UAV nessa direcção. A rotina deverá chamar-se “**Point_to**”, deverá receber um parâmetro inteiro “dir” com a direcção pretendida, e só deve terminar quando o UAV estiver apontado nessa direcção.

I.d) Escreva a rotina principal de controlo da navegação a que chamaremos “**Main_navigation**”. Essa rotina deverá fazer com que o UAV vá andando de um waypoint para o seguinte, e só deverá terminar se a lista de waypoints terminar (com um apontador para NULL). Para passar ao waypoint seguinte, é necessário que a distância entre a posição actual e a posição do waypoint seja inferior à tolerância definida para esse waypoint.

I.e) Escreva uma rotina que preencha a lista de waypoints com os seguintes dados:

```
X= 10  Y=10  
X= 20  Y=20  
X= 40  Y=40  
Fim
```

A rotina deverá devolver um apontador para a lista, que se chamará “first_waypoint”.

I.f) Se quiser que o UAV descreva continuamente um circuito fechado entre os 3 waypoints, o que deverá fazer ?

II

II.a) Explique porque é que pode ser mais eficiente correr dois programas em ambiente de multitasking (i.e., “ao mesmo tempo”) do que correr os mesmos programas sequencialmente num ambiente “single task”.

II.b) Assumindo que haverá um computador “central”, para o sistema de controlo do UAV descrito no grupo I, diga quais são, na sua opinião, as características que o seu sistema operativo deve ter. Em particular discuta as opções (vantagens/desvantagens) quanto a :

Single task / Multitasking

Multitasking preemptivo / Multitasking cooperativo

Mecanismos de exclusão mútua

Gestão de memória (espaço de memória mapeado directamente VS memória virtual, tipo de memória virtual)

Bom trabalho !

