PROBLEMA INTERMEDIÁRIO 1

1 Análise Inicial

O problema pede para encontrar o ângulo ”<ABC”dados três pares de coordenadas dos pontos ”A”, ”B”, e ”C”. Para tal, deciciu-se usar a fórmula do ângulo entre dois vetores com o produto interno, dada por:

$$\cos\theta = \frac{< V_1, V_2 >}{|V_1| \cdot |V_2|} \tag{1}$$

Em que θ é o ângulo entre os vetores V_1 e V_2, $< V_1, V_2 >$ é o produto interno entre os vetores e $|V_1|$ e $|V_2|$ são seus módulos.

2 Estruturas

Como se trata de um problema que envolve o tratamento de coordenadas (x,y) de pontos em um plano cartesiano, fez-se uma estrutura que guarda dois valores *double* para representar essas coordenadas.

```
typedef struct Coord_t{ //Estrutura que guarda as coordenadas (x,y) de um ponto
  double x,y;
} coord;
```

Figura 1: Estrutura de Coordenadas Cartesianas

3 Funções

3.1 Produto Interno

Primeiramente, fez-se a função que calcula e retorna o produto interno entre dois vetores. Ela recebe duas variáveis de coordenadas, que representam os dois vetores e retorna um valor *double* com o valor do produto interno., seguindo a fórmula:

$$< V_1, V_2 >= (V_{1x} \cdot V_{2x}) + (V_{1y} \cdot V_{2y}) \tag{2}$$

Em que V_{ix} e V_{iy} são as coordenadas x e y do vetor i, respectivamente.

```
double prod_int(coord V1, coord V2){ //Retorna o produto interno de dois vetores
  return (V1.x*V2.x) + (V1.y*V2.y);
}
```

Figura 2: Função do Produto Interno de Dois Vetores

3.2 Módulo

Como também precisa-se do valor do módulo de cada vetor, fez-se uma função que calcula isso. Ela recebe as coordenadas de um vetor e retorna seu módulo através da fórmula:

$$|V| = \sqrt{V_x^2 + V_y^2} \tag{3}$$

Em que $|V|$ é o módulo do vetor e V_x e V_y são suas coordenadas x e y.

```
double mod(coord V){ //Retorna o modulo de um vetor
  return sqrt(pow(V.x,2) + pow(V.y,2));
}
```

Figura 3: Função do Módulo de um Vetor

3.3 Ângulo <ABC

Com as funções de produto interno e módulo prontas, fez-se a função que fará o cálculo do ângulo <ABC, dados os pontos A, B e C.

Para tal, primeiro é preciso calcular os vetores que devem estar centrados no ponto B, ou seja, os vetores $\overrightarrow{BA}$ e $\overrightarrow{BC}$. Então, calcula-se da seguinte forma:

$$\overrightarrow{BA} = \overrightarrow{A} - \overrightarrow{B} \tag{4}$$

$$\overrightarrow{BC} = \overrightarrow{C} - \overrightarrow{B} \tag{5}$$

Após isso, usa-se os vetores calculados para calcular o ângulo dado por <ABC a partir da Equação 1, usando as funções de produto interno e módulo apresentadas anteriormente. Com isso, a função retorna o valor do ângulo <ABC em radianos.

```
double angulo_entre_ABC_rad(coord A, coord B, coord C){ //Retorna o angulo
entre tres pontos (x,y) no plano cartesiano em radianos
  coord BA, BC;

  BA.x = A.x - B.x; //Faz se os vetores com centro em B
  BA.y = A.y - B.y;

  BC.x = C.x - B.x;
  BC.y = C.y - B.y;

  return acos(prod_int(BA,BC) / (mod(BA) * mod(BC))); //Retorna o valor do
angulo em radianos usando a formula do produto interno
}
```

Figura 4: Função do Ângulo entre Três Pontos

3.4 Conversão de Ângulos

Como o problema aborda o cálculo de ângulos, decidiu-se também fazer duas funções que calculam a conversão de radianos para graus (rad2deg) e de graus para radianos (deg2rad). Essas funções foram usadas para fazer a verificação dos resultados do programa.

```
double rad2deg(double rad){ //Converte ângulos em radianos para graus
  return rad*180/M_PI;
}

double deg2rad(double deg){ //Converte ângulos em graus para radianos
  return deg*M_PI/180;
}
```

Figura 5: Funções de Conversão entre Graus e Radianos