

## Zadanie 5

---

Definiujemy struktury opisujące punkt na płaszczyźnie

```
struct Point {
    int x, y;
};
```

oraz prostokąt (reprezentowany przez dwa punkty: dolny-lewy i górny-prawy wierzchołek)

```
struct Rect {
    Point bottom_left, top_right;
};
```

Napisz i przetestuj funkcje:

```
void printPoint(const Point& p);
void printRect(const Rect* r);
bool doIntersect(const Rect* r1, const Rect* r2);
Rect intersection(const Rect& r1, const Rect& r2);
```

(zauważ, że używane są tu zarówno wskaźniki jak i referencje) o następującej funkcjonalności:

- **printPoint** — wypisuje informacje o punkcie;
- **printRect** — wypisuje informacje o prostokącie; może korzystać z **printPoint**);
- **doIntersect** — sprawdza, czy dwa podane prostokąty mają część wspólną;
- **intersection** — zwraca przez wartość prostokąt będący częścią wspólną (przecięciem) dwóch podanych prostokątów, lub zgłasza wyjątek, jeśli podane dwa prostokąty nie mają części wspólnej (np. `throw "Error in 'intersection'";`).

Na przykład, funkcja **main**

```
int main() {
    Rect r1{{1,1},{4,3}};
    Rect r2{{2,0},{3,4}};
    Rect r3{{0,4},{1,5}};

    if (haveIntersection(&r1,&r2)) {
        Rect r = intersection(r1,r2);
        printRect(&r);
    } else
```

```

        cout << "No intersection" << endl;

    if (haveIntersection(&r1,&r3)) {
        Rect r = intersection(r1,r3);
        printRect(&r);
    } else
        cout << "No intersection" << endl;
}

```

powinna wypisać

```

[(2,1),(3,3)]
No intersection

```

**UWAGA:**

Algorytm pozwalający znaleźć przecięcie jest bardzo prosty: położenie lewej krawędzi przecięcia to maksimum z położen lewych krawędzi prostokątów, położenie prawej krawędzi przecięcia — minimum z położen prawych krawędzi. Jeśli lewa wypadnie na prawo od prawej, to przecięcie nie istnieje. Podobnie dla krawędzi górnej i dolnej.

*Termin: do 14 grudnia (włącznie)*

Rozwiązania, w postaci **jednego** pliku źródłowego zawierającego treść programu, proszę wrzucać w systemie EDU do katalogu „Foldery zadań / Zadanie\_XX”, gdzie 'XX' jest numerem zadania.

Nazwą pliku powinno być nazwisko z dużej litery (bez polskich znaków); rozszerzeniem musi być '.cpp', czyli np. *Malinowska.cpp*.