BPC-UDP Cvičení 08

# Příklad 1

Napište a odlaďte program, který přečte ze standardního vstupu souřadnice bodu v rovině a vypočítá vzdálenost bodu od počátku souřadnic a tuto vzdálenost vytiskne na standardní výstup.

1. Napište definici funkce distance(), která vypočte vzdálenost bodu od počátku souřadnic a vrátí ji jako návratovou hodnotu typu **float**. Funkce bude mít 2 parametry: aX a aY typu **float**. (Vzdálenost bodu od počátku souřadnic je dána vztahem ).
2. Uveďte deklaraci (úplný funkční prototyp) funkce distance().
3. Ve funkci main():
   1. Definujte proměnné x a y typu **float**. Ze standardního vstupu přečtete hodnoty souřadnic bodu a uložte je do proměnných x a y. Pokud při čtení souřadnic dojde k chybě, program ukončete s návratovým kódem 1.
   2. Z funkce main() zavolejte funkci distance() a předejte ji souřadnice x a y jako argumenty. Návratovou hodnotu funkce uložte do proměnné r typu **float**.
   3. Vytiskněte hodnotu vzdálenosti bodu od počátku souřadnic na standardní výstup.

Např. pro souřadnice [3, 4] vyjde vzdálenost bodu 5

# Příklad 2

Napište program, který přečte ze standardního vstupu souřadnice bodu v rovině a uloží je do objektu datového typu struktura TPoint. Následně vypočítá vzdálenost bodu od počátku souřadnic a tuto vzdálenost vytiskne na standardní výstup.

1. Definujte datový typ struktura TPoint, který bude mít členy iX a iY oba typu **float** (iX a iY reprezentující souřadnice bodu v rovině).
2. Ve funkci main():
   1. Definujte proměnnou point typu struktura TPoint a inicializujte ji v definici tak, aby členy iX a iY měly hodnotu 0.
   2. Ze standardního vstupu přečtěte souřadnice bodu a uložte je do členů iX a iY proměnné point. Pokud při čtení souřadnic dojde k chybě, program ukončete s návratovým kódem 1.
   3. Definujte proměnnou r typu float. Napište výraz, který do proměnné r uloží vzdálenost bodu od počátku souřadnic.
   4. Vytiskněte vypočtenou vzdálenost bodu na standardní výstup.

# Příklad 3

Upravte programy z příkladu 2 a 3 tak, aby souřadnice bodu v rovině byly čteny ve funkci read\_point() a vzdálenost bodu byla počítána ve funkci distance().

1. Napište deklaraci (úplný funkční prototyp) funkce read\_point(). Funkce nebude mít žádný parametr. Návratová hodnota funkce bude typu struktura TPoint.
2. Napište deklaraci (úplný funkční prototyp) funkce distance(). Funkce bude mít formální parametr aPoint typu struktura TPoint. Návratová hodnota funkce bude typu **float**.
3. Ve funkci main():
   1. Definujte proměnnou point typu struktura TPoint a inicializujte ji v definici tak, aby členy iX a iY měly hodnotu 0.
   2. Zavolejte funkci read\_point(). Návratovou hodnotu uložte do proměnné point.
   3. Pokud mají oba členy proměnné point hodnotu 0, program ukončete s návratovým kódem 1.
   4. Z funkce main() zavolejte funkci distance() a předejte ji proměnnou point jako argument.
   5. Vytiskněte hodnotu vzdálenosti bodu od počátku souřadnic (návratovou hodnotu funkce distance()) na standardní výstup.
4. Napište definici funkce read\_point(), která ze standardního vstupu přečtěte souřadnice bodu a vrátí je prostřednictvím návratové hodnoty typu struktura TPoint. Pokud při čtení souřadnic dojde k chybě, funkce vrátí objekt typu struktura TPoint s nulovými hodnotami složek iX a iY.
5. Napište definici funkce distance(), která vypočítá vzdálenost bodu od počátku souřadnic a vrátí ji jako návratovou hodnotu.

# Příklad 4

Napište a odlaďte program, který ze standardního vstupu přečte souřadnice levého horního rohu a pravého dolního rohu obdélníku, stanoví délku úhlopříčky a vytiskne ji na standardní výstup.

1. Definujte datový typ struktura TPoint, který bude mít členy iX a iY oba typu **float** (iX a iY reprezentující souřadnice bodu v rovině).
2. Definujte datový typ struktura TRect, který bude mít členy iPointLH a iPointRD, oba typu TPoint (iPointLH reprezentuje horní levý roh a iPointRD reprezentuje pravý dolní roh).
3. Ve funkci main():
   1. Definujte proměnnou rectangle typu struktura TRect a inicializujte ji v definici tak, aby souřadnice rohů obdélníku měly hodnotu 0.
   2. Vypište na standardní výstup výzvu uživateli "Zadej souradnice leveho horniho rohu:".
   3. Ze standardního vstupu přečtěte souřadnice levého horního rohu a uložte je do složky iPointLH proměnné rectangle.
   4. Vypište na standardní výstup výzvu uživateli "Zadej souradnice praveho dolniho rohu:"
   5. Ze standardního vstupu přečtěte souřadnice pravého dolního rohu a uložte je do složky iPointRD proměnné rectangle.
   6. Definujte proměnnou diag typu **float**. Napište výraz, který do proměnné diag uloží délku úhlopříčky.
   7. Vytiskněte vypočtenou délku úhlopříčky na standardní výstup.

Např. pro souřadnice [1, 6] a [4, 2] vyjde délka úhlopříčky 5.

# Příklady pro samostatné procvičení probrané látky

# Příklad 5

Upravte program z příkladu 4 tak, aby souřadnice rohů obdélníku byly čteny pomocí funkce read\_rect(). Funkce bude využívat funkci read\_point(). Délka uhlopříčky stanovte pomocí funkce diagonal() definované jako **static inline**.

1. Napište definici funkce read\_rect()Funkce nebude mít žádný parametr. Návratová hodnota funkce bude typu struktura TRect.
   1. Definujte proměnnou rect typu struktura TRect a inicializujte ji v definici tak, aby souřadnice rohů obdélníku měly hodnotu 0.
   2. Vypište na standardní výstup výzvu uživateli "Zadej souradnice leveho horniho rohu:".
   3. Pomocí funkce read\_point() přečtěte souřadnice levého horního rohu a uložte je do složky iPointLH proměnné rect.
   4. Vypište na standardní výstup výzvu uživateli "Zadej souradnice praveho dolniho rohu:".
   5. Pomocí funkce read\_point() přečtěte souřadnice pravého dolního rohu a uložte je do složky iPointRD proměnné rect.
   6. Vraťte hodnotu proměnné rect a funkci ukončete.
2. Napište deklaraci (úplný funkční prototyp) funkce read\_rect().
3. Napište definici funkce **static inline** funkce diagonal(). Funkce vypočítá délku diagonály obdélníka a vrátí ji jako návratovou hodnotu.
4. Upravte funkci main() tak, aby pro čtení souřadnic rohů obdélníku využila funkci read\_rect() a pro výpočet délky uhlopříčky funkci diagonal().