

Může být nutné nainstalovat následující knihovny:

```
libsigc++-2.0-dev
```

```
libgtkmm-3.0-dev
```

Rámcový postup řešení:

planner.cc:

Načtení mapy

Ve funkci `setMap()` si uložte mapu a případně rovnou zavolejte nafouknutí překážek funkcí `inflateMap()`. Tu naimplementujte podle přednášky. Uvažujte velikost buňky 2cm a robot, jehož průměr je 18cm. Pro uložení mapy si můžete vytvořit vlastní struktury zahrnující například značení uzlů mapy či graf sousedností.

Dijkstra

Ve funkci `dijkstra()` implementujte Dijkstrův algoritmus pro nejkratší cestu v grafu s použitím [prioritní fronty STL](#). Můžete upravit návratovou hodnotu i parametry, nebo výsledky ukládat do vlastní třídní proměnné.

Plánování cesty

Z funkce `plan()` zavolejte Dijkstrův algoritmus a nalezněte nejkratší cestu mezi ohodnocenými uzly grafu. Dále zavolejte funkci `smooth_path()` pro vyhlazení trasy. Ve funkci `smooth_path()` postupujte podle přednášek, můžete využít funkci `bresenham()`, která ověří viditelnost mezi dvěma body. Pro vykreslení mapy si upravte návratovou hodnotu na nalezenou cestu.

main.cc:

Vykreslení cesty

Z funkce `planning()` zavolejte po inicializaci mapy vámi implementované funkce plánovače `setMap()` a `plan()`. Výslednou trasu můžete vykreslit s pomocí vzorových funkcí "drawXX".