

## Instructions for translators

1. Open this file on GitHub server. If you see `https://um.mendelu.cz/...` in URL, click View on GitHub to open this file on github.com.
2. If you see this file on GitHub server, you can edit the content of the file. Open the file in an editor. You can use simple editor (press e on GitHub). However, an advanced VS Code editor (press . on GitHub) is better, since it provides preview how the Markdown code renders. Alternatively press pencil for simple editor or press triangle next to the pencil to get access to VS Code described as github.dev.
3. Fix the keywords in the preamble.
4. Depending on which language version you want to use as a source for your translation, delete either English or Czech version below.
5. Translate to your language. Keep Markdown marking and math notation. If you use a tool to get first version of the translation, make sure that the markup is preserved.
6. In VS Code you can open the preview in another window by pressing `Ctrl+V` and `K`. Keep the preview open as you work, or close using a mouse.
7. Instead of saving, you have to commit and push the changes to the repository. Fill the Message under Source control (describe your changes, such as “Polish translation started”) and then press Commit&Push.
8. Make sure that your changes appear in the commit history. In rare cases (if you work with simultaneously with someone else) you have to download /Pull/ and merge his and yours changes. Usually Sync (Pull & Push) should work.
9. When you finish the translation, change `is_finished: False` in header to `is_finished: True`.

## Instrukce pro překladatele

1. Otevřete tento soubor na serveru GitHub. Pokud máte soubor otevřen na `https://um.mendelu.cz/...`, otevřete jej na serveru github.com.
2. Pokud tento soubor vidíte na serveru GitHub, můžete obsah souboru upravit. Otevřete soubor v editoru. Můžete použít jednoduchý editor (stiskněte e na GitHubu). Lepší je však pokročilý editor VS Code (stikněte . na GitHubu), protože poskytuje náhled, jak se kód Markdown interpretuje. Případně stiskněte tužku pro jednoduchý editor nebo stiskněte trojúhelníček vedle tužky, abyste získali přístup k editoru VS Code popsaný jako github.dev.
3. Opravte klíčová slova v preambuli.
4. V závislosti na tom, kterou jazykovou verzi chcete použít jako zdrojový kód pro svůj překlad, odstraňte níže uvedenou anglickou nebo českou verzi.
5. Přeložte do svého jazyka. Ponechte značení Markdown a matematický zápis. Pokud použijete nástroj typu DeepL pro získání první verze překladu, ujistěte se, že zápis matematických výrazů byl zachován.
6. Ve VS Code můžete náhled otevřít v jiném okně stisknutím `Ctrl+V`. a `K`. Během práce nechte náhled otevřený nebo jej zavřete pomocí myši.
7. Místo uložení musíte změny zaregistrovat a odeslat do úložiště. Vyplňte zprávu v poli Zpráva (popište své změny, např. “Zahájen překlad do polštiny”) a poté stiskněte tlačítko Commit&Push.

8. Ujistěte se, že se vaše změny objeví v historii revizí. Ve výjimečných případech (pokud pracujete současně s někým jiným) musíte stáhnout /Pull/ a sloučit jeho a vaše změny. Obvykle by synchronizace (Pull & Push) měla fungovat.
9. Po dokončení překladu změňte `is_finished: False` v záhlaví na `is_finished: True`.

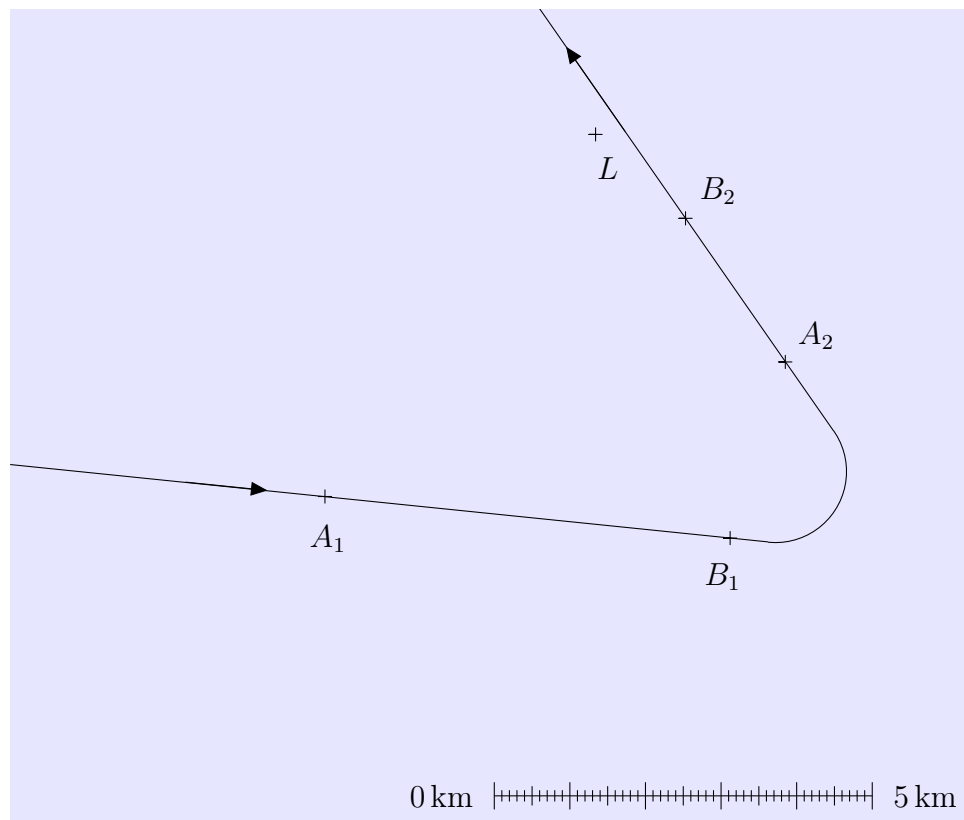
## Czech source

# Záchrana trosečníka

*Keywords: plane geometry, Pythagorean theorem, perpendicular bisector*

Letadlo hledá na otevřeném moři pozici trosečníka, který má na svém voru zařízení vysílající tísňový signál. Zařízení má pouze omezený dosah. Při letu nad mořem posádka letadla signál zachytí, ale po chvíli jej ztratí. Pilot proto letadlo stočí a podaří se mu signál na kratší dobu opět zachytit.

Trajektorie celého letu je i s naznačeným směrem a místy zachycení (body  $A_1$  a  $A_2$ ) a ztrát (body  $B_1$  a  $B_2$ ) signálu znázorněn na mapě.



**Rysunek 1:** Trajektorie letu letadla

Během obou dob, kdy posádka přijímala signál, letadlo neměnilo svou výšku, mezi body  $B_1$  a  $A_2$  snížilo

Results matter!

svou výšku o 500 m.

**Úloha 1.** Konstrukčně určete v mapě pozici  $X$  trosečníka.

**Úloha 2.** V lokalitě se nachází dopravní loď (pozice  $L$ ). Může také zaznamenat trosečnickův tísňový signál, nebo je příliš daleko?

- Velikosti úseček  $LX$  a  $A_1X$ ,  $A_2X$  z řešení Úlohy 1 přeneste na měřítko. Užitím takto určených vzdáleností (zaokrouhlených na celý nejmenší dílek stupnice) vyřešte úlohu počtně.
- Vyjděte z řešení Úlohy 1 a úlohu vyřešte znovu, tentokrát pouze užitím geometrických konstrukcí.

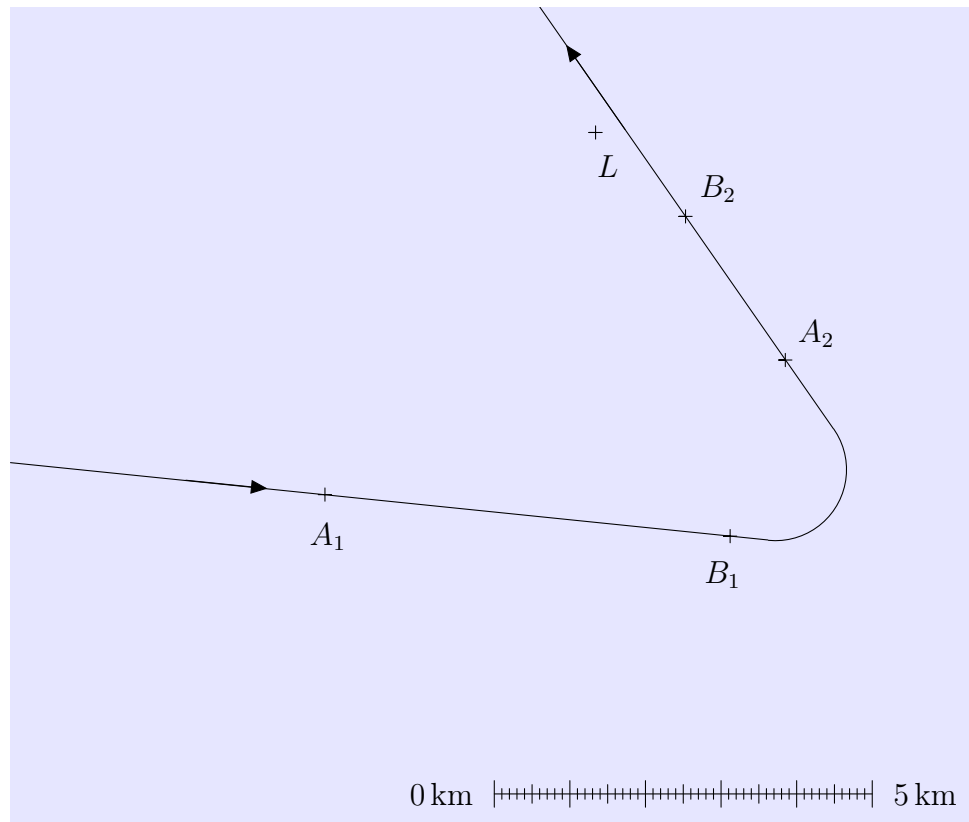
## English source

# Rescuing a Castaway

*Keywords: plane geometry, Pythagorean theorem, perpendicular bisector*

A plane is searching the open sea for a castaway who has a device on their raft that emits a distress signal. The device has only a limited range. While flying over the sea, the crew picks up the signal, but after a short time, it is lost. The pilot turns the aircraft around, and they manage to receive the signal again, though only briefly.

The trajectory of the entire flight, including the direction of travel and the points where the signal was picked up (points  $A_1$  a  $A_2$ ) and lost (points  $B_1$  and  $B_2$ ) is shown on the map.



Rysunek 2: Flight trajectory of the aircraft

During both periods when the crew received the signal, the aircraft maintained a constant altitude. Between points  $B_1$  and  $A_2$ , the aircraft descended by 500, m.

**Exercise 1.** Use a geometric construction on the map to determine the position  $X$  of the castaway.

**Exercise 2.** There is a cargo ship in the area (position  $L$ ). Can it also receive the castaway's distress signal, or is it too far away?

- a) Transfer the lengths of segments  $LX$ ,  $A_1X$ , and  $A_2X$  from the solution to Exercise 1 to the scale provided. Using these distances (rounded to the nearest smallest unit of the scale), solve the problem numerically.
- b) Using the construction from Exercise 1, solve the problem again—this time relying solely on geometric constructions.