

Stvaranje SFML projekta u Visual Studiu

Objektno programiranje - 8. vježbe (2. dodatak)

Sebastijan Horvat

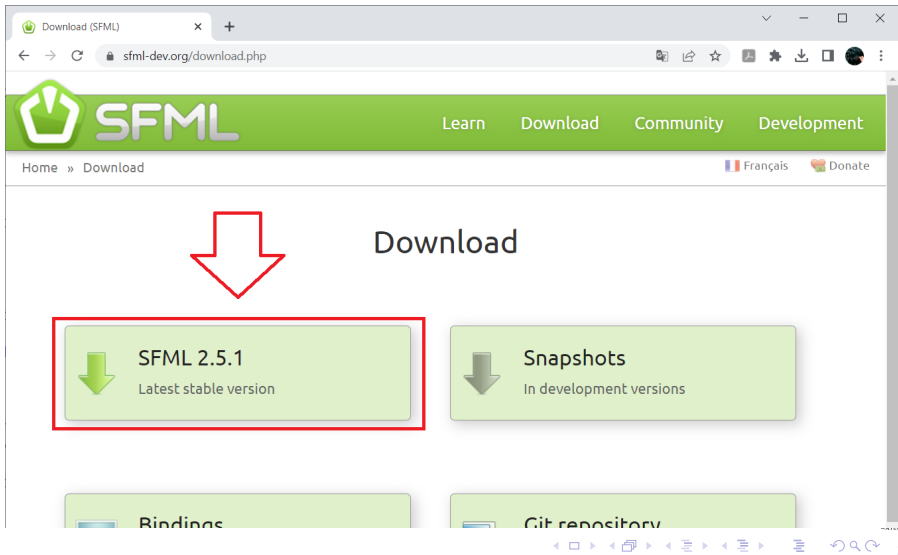
Prirodoslovno-matematički fakultet,
Sveučilište u Zagrebu

10. svibnja 2023. godine



Preuzimanje SFML-a

www.sfml-dev.org/download.php



The screenshot shows a web browser window with the address bar displaying "sfml-dev.org/download.php". The page features a green header with the SFML logo and navigation links for "Learn", "Download", "Community", and "Development". Below the header, there is a breadcrumb trail "Home » Download" and a language selector for "Français" and a "Donate" button. The main content area is titled "Download" and contains several green buttons. A large red arrow points down to the "SFML 2.5.1" button, which is also highlighted with a red rectangular border. The "SFML 2.5.1" button includes a green download icon and the text "Latest stable version". Other buttons visible include "Snapshots" (In development versions), "Bindings", and "Git repository".

Download (SFML) x +

sfml-dev.org/download.php

SFML Learn Download Community Development

Home » Download Français Donate

Download

↓

↓ SFML 2.5.1
Latest stable version

↓ Snapshots
In development versions

Bindings Git repository



Download SFML 2.5.1

On Windows, choosing 32 or 64-bit libraries should be based on which platform you want to compile for, not which OS you have. Indeed, you can perfectly compile and run a 32-bit program on a 64-bit Windows. So you'll most likely want to target 32-bit platforms, to have the largest possible audience. Choose 64-bit packages only if you have good reasons.

Unless you are using a newer version of Visual Studio, the compiler versions have to match 100%!

In case you are using Visual Studio 2015 or newer, you can go ahead and use the SFML version compiled for Visual C++ 15 (VS 2017), which is still compatible with newer versions of the toolchain.

Here are links to the specific MinGW compiler versions used to build the provided packages:

TDM 5.1.0 (32-bit), MinGW Builds 7.3.0 (32-bit), MinGW Builds 7.3.0 (64-bit)

Visual C++ 15 (2017) - 32-bit

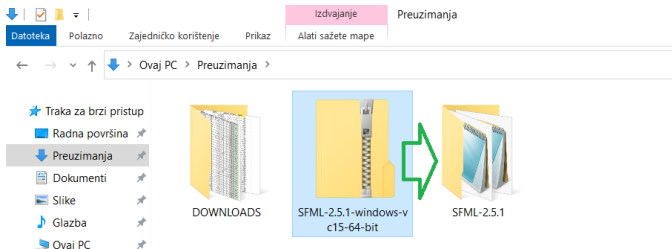
[Download](#) | 16.3 MB

Visual C++ 15 (2017) - 64-bit

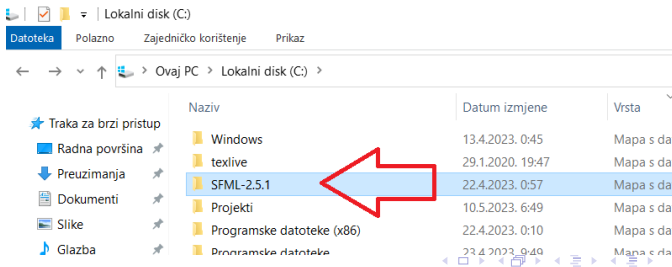
[Download](#) | 18.0 MB

Visual C++ 14 (2015) - 64-bit

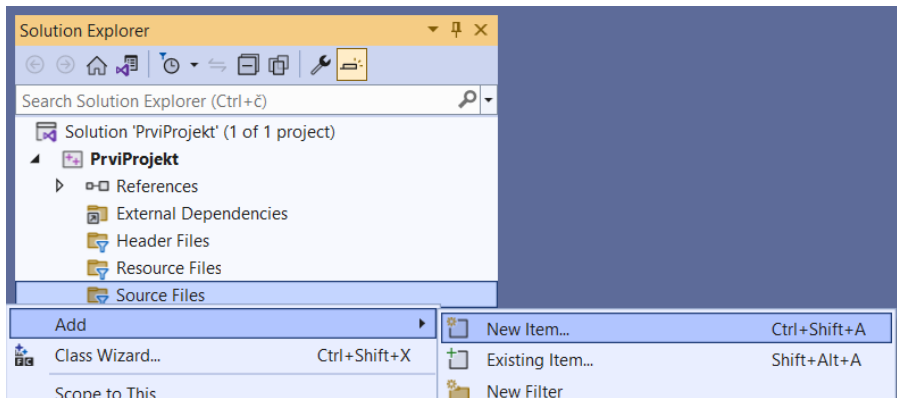
Unzipati i spremite mapu



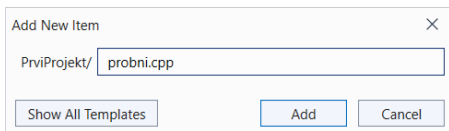
- mapu spremite tamo gdje ćemo ju kasnije lako naći!
(ako je kao na slici, adresa je: **C:\SFML-2.5.1**)



Dodavanje koda u projekt (Visual Studio 2022)



- gornja slika: desni klik na *Source Files* (→ *Add* → *New Item...*)

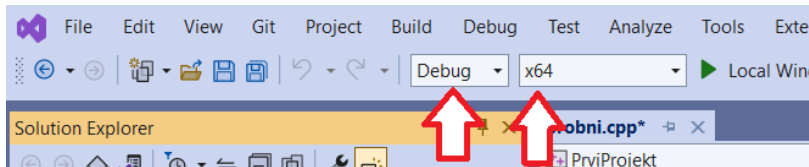


Kod za testiranje (može se preuzeti na stranici kolegija)

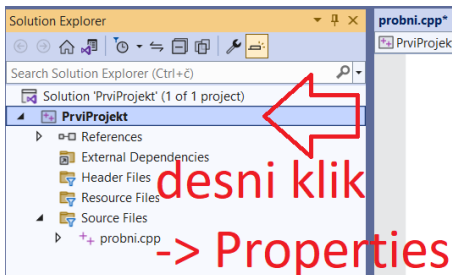
```
1  #include <SFML/Graphics.hpp>
2
3  int main() {
4      sf::RenderWindow window(sf::VideoMode(640, 480), "Prvi prozor!");
5      sf::CircleShape shape(150.f);
6      shape.setFillColor(sf::Color::Blue);
7      while (window.isOpen()) {
8          sf::Event event;
9          while (window.pollEvent(event)) {
10             if (event.type == sf::Event::Closed)
11                 window.close();
12         }
13         window.clear();
14         window.draw(shape);
15         window.display();
16     }
17     return 0;
18 }
```

Postavljanje svojstava projekta

- mora biti odabrano *Debug* i *x64* (kao na slici dolje)



- desni klik na naziv projekta (kao na slici dolje) → *Properties*



Postavljanje svojstava projekta

- treba reći kompajleru gdje naći SFML zaglavlja (.hpp datoteke)
- C/C++ → *General* → *Additional Include Directories* - upisati *gdje-instalirali-SFML/include* - u ovom slučaju:
C:\SFML-2.5.1\include
- putanje su iste i za *Debug* i za *Release* konfiguraciju pa ih postavljamo globalno (*All configurations*)

PrviProjekt Property Pages

The screenshot shows the Visual Studio Property Pages window for a project named 'PrviProjekt'. The 'Configuration' dropdown is set to 'All Configurations' and the 'Platform' is 'Active(x64)'. In the left-hand tree view, 'C/C++' is expanded, and 'General' is selected. The main pane shows the 'Additional Include Directories' property set to 'C:\SFML-2.5.1\include'. Other properties like 'Additional #using Directories', 'Additional BMI Directories', and 'Additional Module Dependencies' are also visible.

Property	Value
Configuration	All Configurations
Platform	Active(x64)
Additional Include Directories	C:\SFML-2.5.1\include
Additional #using Directories	
Additional BMI Directories	
Additional Module Dependencies	
Additional Header Unit Dependencies	
Scan Sources for Module Dependencies	No
Translate Includes to Imports	No
Debug Information Format	<different options>

Postavljanje svojstava projekta

- treba reći linkeru gdje naći SFML biblioteke (.lib datoteke)
- *Linker* → *General* → *Additional Library Directories* - upisati gdje instalirali SFML/lib - u ovom slučaju:
C:\SFML-2.5.1\lib

PrviProjekt Property Pages

Configuration: All Configurations Platform: Active(x64)

Configuration Properties

- General
- Advanced
- Debugging
- VC++ Directories
- C/C++
 - Linker**
 - General**
 - Input
 - Manifest File
 - Debugging
 - System

Output File	\$(OutDir)\$(TargetName)\$(TargetExt)
Show Progress	Not Set
Version	
Enable Incremental Linking	<different options>
Incremental Link Database File	\$(IntDir)\$(TargetName).ilk
Suppress Startup Banner	Yes (/NOLOGO)
Ignore Import Library	No
Register Output	No
Per-user Redirection	No
Additional Library Directories	C:\SFML-2.5.1\lib
Link Library Dependencies	Yes
Use Library Dependency Inputs	No

Linkanje potrebnih SFML biblioteka

- treba linkati SFML biblioteke (.lib datoteke) koje će naš kod trebati (uz dodatak -d u nazivu datoteke za *Debug*)
- SFML se sastoji od **5 modula**: *system*, *window*, *graphics*, *network* i *audio*

PrviProjekt Property Pages

Configuration: All Configurations Platform: Active(x64) Configuration Manager...

Configuration Properties

- General
- Advanced
- Debugging
- VC++ Directories
- C/C++
 - Linker
 - General
 - Input

Additional Dependencies: \$(CoreLibraryDependencies);%(AdditionalDependencies)

Ignore All Default Libraries <Edit...>

Ignore Specific Default Libraries

Module Definition File

Add Module to Assembly

Embed Managed Resource File

Force Symbol References

Delay Loaded DLLs

Assembly Link Resource

- *Linker* → *Input* → *Additional Dependencies* (slika gore)
- navedemo sve koje trebamo (slika desno)

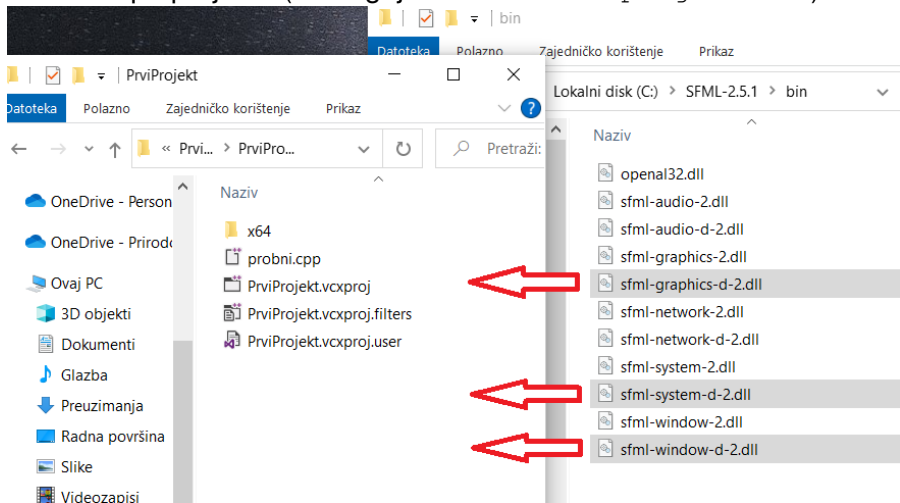
Additional Dependencies

```
sfml-graphics-d.lib  
sfml-window-d.lib  
sfml-system-d.lib
```



Posljednja stvar

- prebaciti potrebne `.dll` datoteke (iz *gdje-instalirali-SFML/bin*) u mapu projekta (tamo gdje se nalazi `.vcxproj` datoteka)



(Razmislite o YOLO pristupu: sve uključiti i sve prebaciti.)

Pokretanje (ako prije sve dobro postavljeno)

