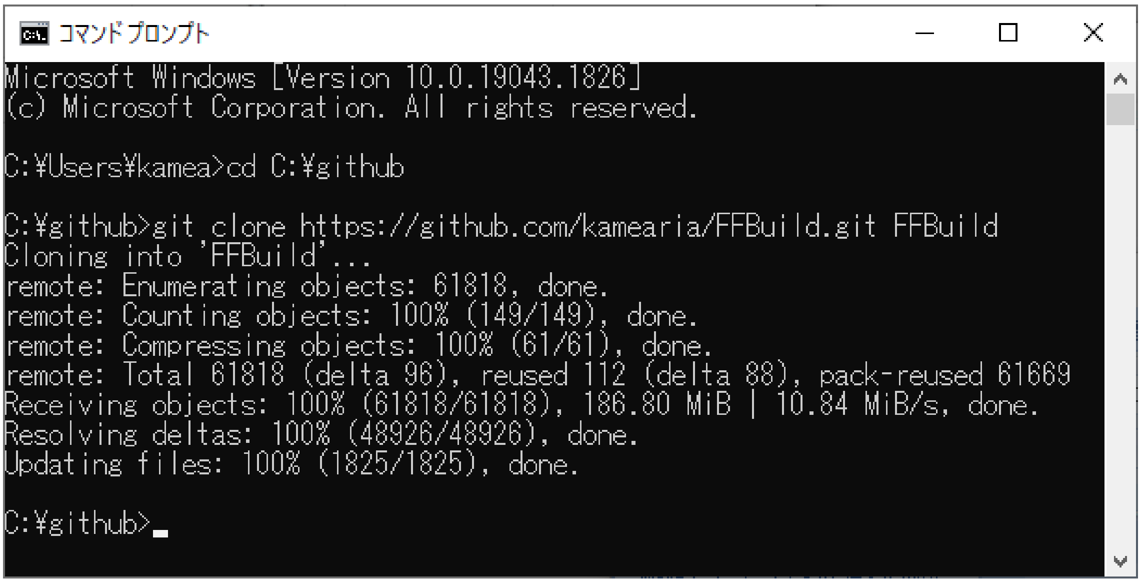
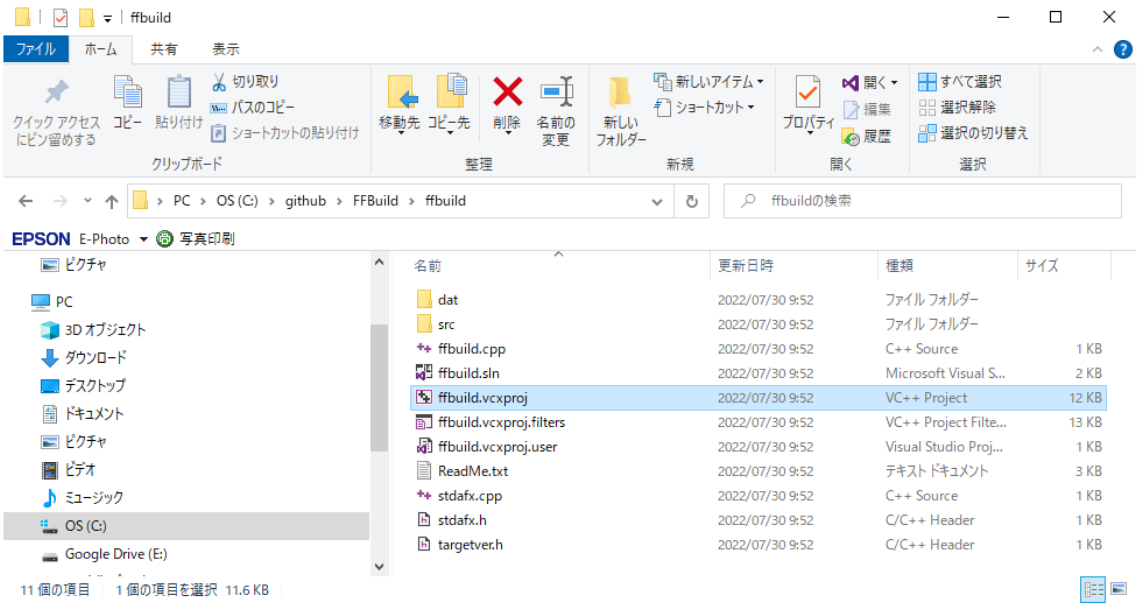
１．Github の repository をクローン



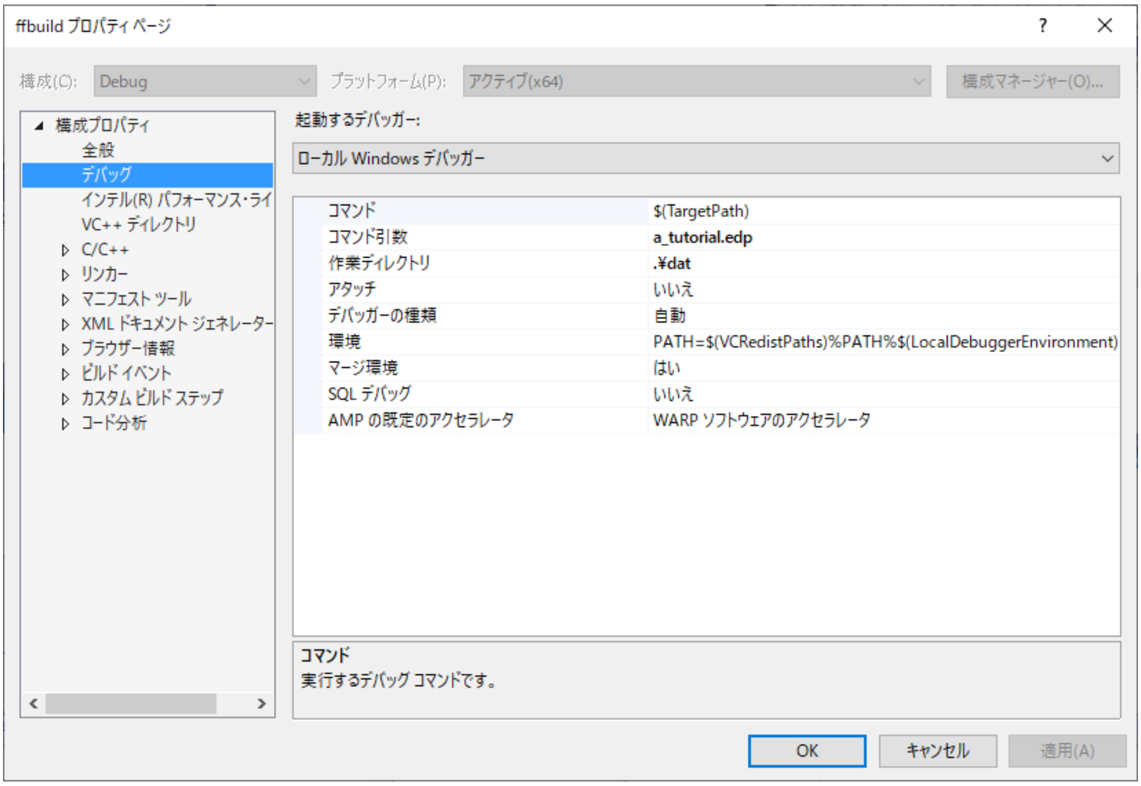
```
Microsoft Windows [Version 10.0.19043.1826]
(c) Microsoft Corporation. All rights reserved.

C:¥Users¥kamea>cd C:¥github

C:¥github>git clone https://github.com/kamearia/FFBuild.git FFBuild
Cloning into 'FFBuild'...
remote: Enumerating objects: 61818, done.
remote: Counting objects: 100% (149/149), done.
remote: Compressing objects: 100% (61/61), done.
remote: Total 61818 (delta 96), reused 112 (delta 88), pack-reused 61669
Receiving objects: 100% (61818/61818), 186.80 MiB | 10.84 MiB/s, done.
Resolving deltas: 100% (48926/48926), done.
Updating files: 100% (1825/1825), done.

C:¥github>_
```

２．Visual Studio を立ち上げる。



名前	更新日時	種類	サイズ
dat	2022/07/30 9:52	ファイル フォルダー	
src	2022/07/30 9:52	ファイル フォルダー	
ffbuild.cpp	2022/07/30 9:52	C++ Source	1 KB
ffbuild.sln	2022/07/30 9:52	Microsoft Visual S...	2 KB
ffbuild.vcxproj	2022/07/30 9:52	VC++ Project	12 KB
ffbuild.vcxproj.filters	2022/07/30 9:52	VC++ Project Filte...	13 KB
ffbuild.vcxproj.user	2022/07/30 9:52	Visual Studio Proj...	1 KB
ReadMe.txt	2022/07/30 9:52	テキスト ドキュメント	3 KB
stdafx.cpp	2022/07/30 9:52	C++ Source	1 KB
stdafx.h	2022/07/30 9:52	C/C++ Header	1 KB
targetver.h	2022/07/30 9:52	C/C++ Header	1 KB

３．デバッガーの設定（設定済み）

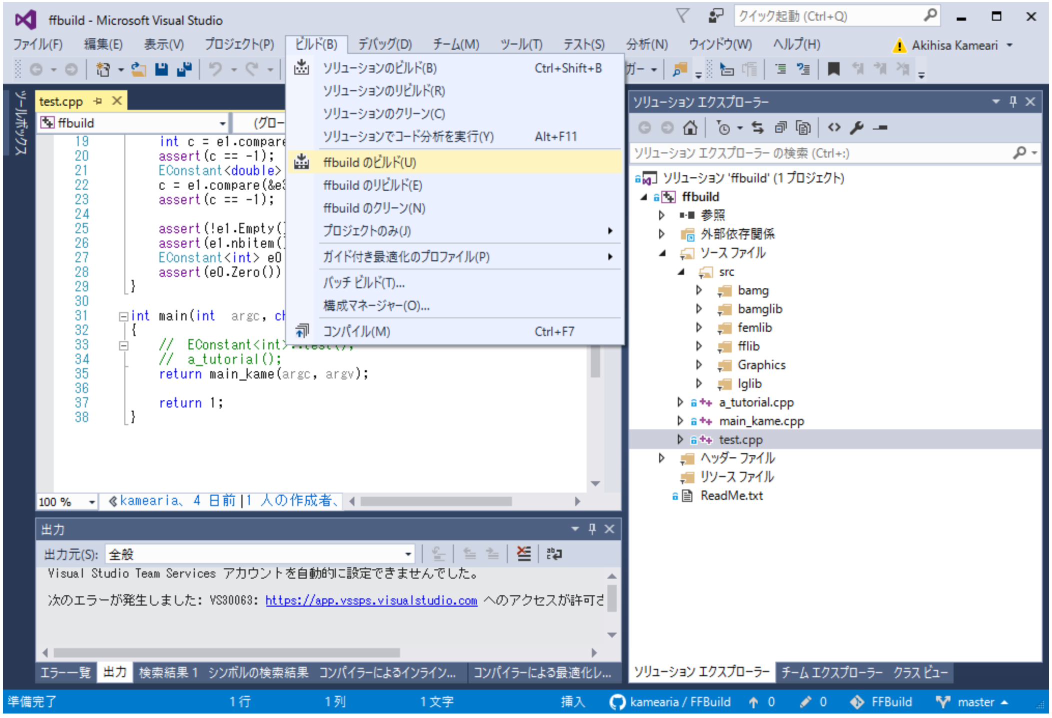


起動するデバッガー: ローカル Windows デバッガー

コマンド	$(TargetPath)
コマンド引数	**a_tutorial.edp**
作業ディレクトリ	**.¥dat**
アタッチ	いいえ
デバッガーの種類	自動
環境	PATH=$(VCRedistPaths)%PATH%$(LocalDebuggerEnvironment)
マージ環境	はい
SQL デバッグ	いいえ
AMP の既定のアクセラレータ	WARP ソフトウェアのアクセラレータ

コマンド
実行するデバッグ コマンドです。

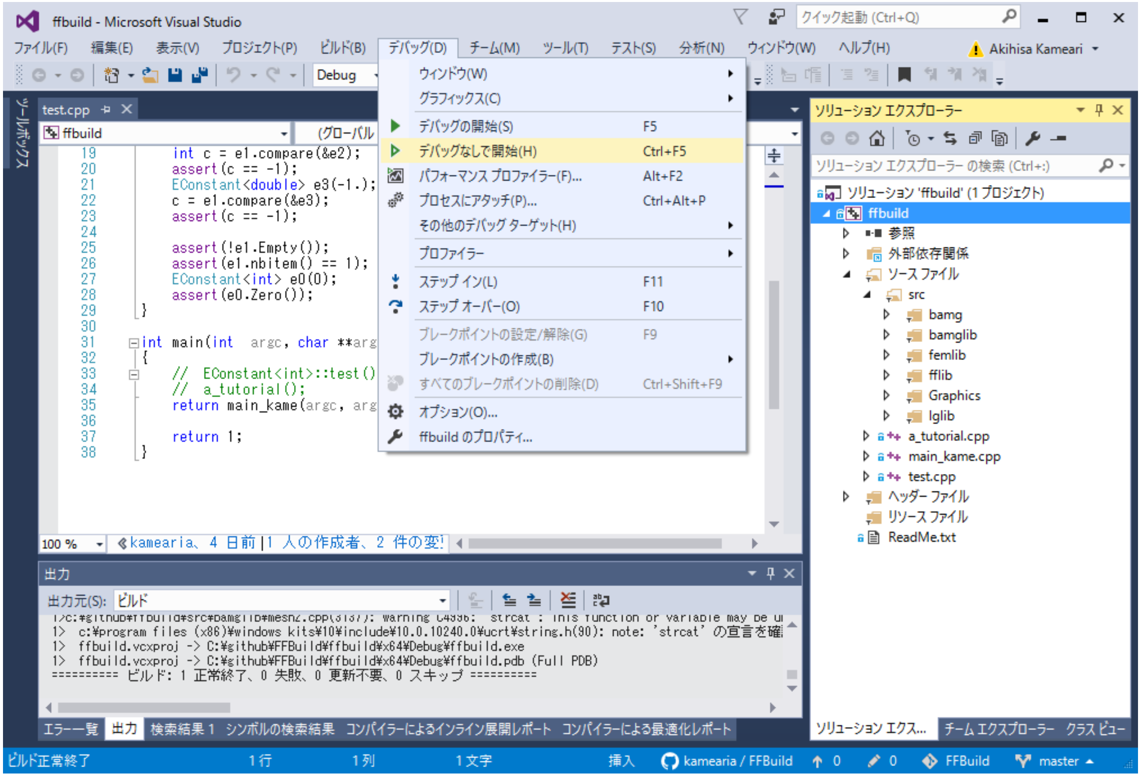
４．FFbuild のビルド



```
    int c = e1.compare
    assert(c == -1);
    EConstant<double>
    c = e1.compare(&e3
    assert(c == -1);

    assert(!e1.Empty(
    assert(e1.nbitem(
    EConstant<int> e0
    assert(e0.Zero())
}

int main(int  argc, ch
{
    //  EConstant<int>::test();
    //  a_tutorial();
    return main_kame(argc, argv);

    return 1;
}
```

```
Visual Studio Team Services アカウントを自動的に設定できませんでした。

次のエラーが発生しました: VS30063: https://app.vssps.visualstudio.com へのアクセスが許可さ
```

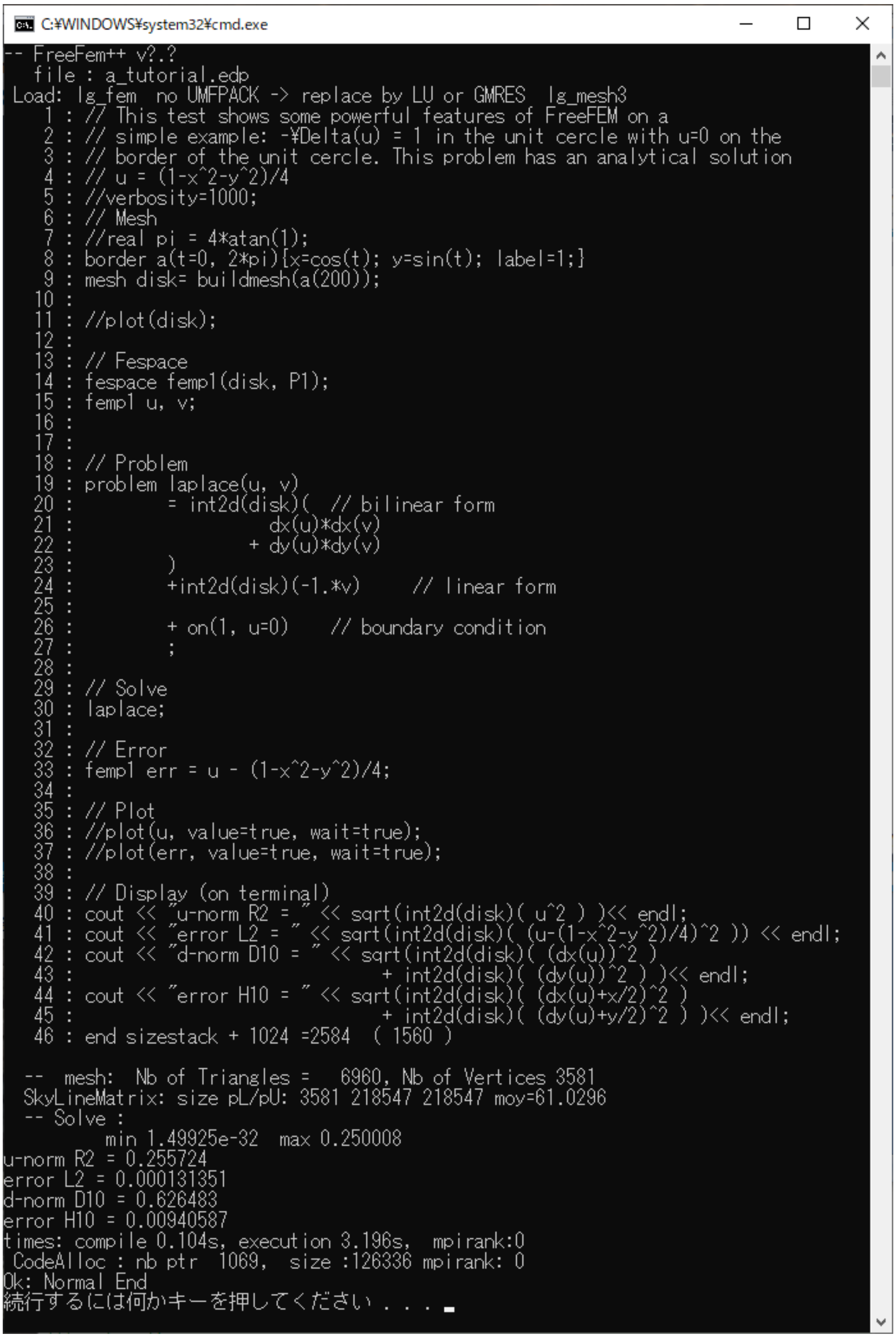
５．デバッグなしで開始



```
        int c = e1.compare(&e2);
        assert(c == -1);
        EConstant<double> e3(-1.);
        c = e1.compare(&e3);
        assert(c == -1);

        assert(!e1.Empty());
        assert(e1.nbitem() == 1);
        EConstant<int> e0(0);
        assert(e0.Zero());
    }

int main(int  argc, char **arg
{
    //  EConstant<int>::test()
    //  a_tutorial();
    return main_kame(argc, arg

    return 1;
}
```

```
1>c:¥github¥ffbuild¥src¥bamglib¥mesh2.cpp(3137): warning C4996: 'strcat': This function or variable may be ur
1>  c:¥program files (x86)¥windows kits¥10¥include¥10.0.10240.0¥ucrt¥string.h(90): note: 'strcat' の宣言を確
1>  ffbuild.vcxproj -> C:¥github¥FFBuild¥ffbuild¥x64¥Debug¥ffbuild.exe
1>  ffbuild.vcxproj -> C:¥github¥FFBuild¥ffbuild¥x64¥Debug¥ffbuild.pdb (Full PDB)
========== ビルド: 1 正常終了、0 失敗、0 更新不要、0 スキップ ==========
```

６．出力



```
-- FreeFem++ v?.?
  file : a_tutorial.edp
 Load: lg_fem  no UMFPACK -> replace by LU or GMRES  lg_mesh3
    1 : // This test shows some powerful features of FreeFEM on a
    2 : // simple example: -¥Delta(u) = 1 in the unit cercle with u=0 on the
    3 : // border of the unit cercle. This problem has an analytical solution
    4 : // u = (1-x^2-y^2)/4
    5 : //verbosity=1000;
    6 : // Mesh
    7 : //real pi = 4*atan(1);
    8 : border a(t=0, 2*pi){x=cos(t); y=sin(t); label=1;}
    9 : mesh disk= buildmesh(a(200));
   10 :
   11 : //plot(disk);
   12 :
   13 : // Fespace
   14 : fespace femp1(disk, P1);
   15 : femp1 u, v;
   16 :
   17 :
   18 : // Problem
   19 : problem laplace(u, v)
   20 :         = int2d(disk)(  // bilinear form
   21 :                  dx(u)*dx(v)
   22 :                + dy(u)*dy(v)
   23 :         )
   24 :         +int2d(disk)(-1.*v)     // linear form
   25 :
   26 :         + on(1, u=0)    // boundary condition
   27 :         ;
   28 :
   29 : // Solve
   30 : laplace;
   31 :
   32 : // Error
   33 : femp1 err = u - (1-x^2-y^2)/4;
   34 :
   35 : // Plot
   36 : //plot(u, value=true, wait=true);
   37 : //plot(err, value=true, wait=true);
   38 :
   39 : // Display (on terminal)
   40 : cout << "u-norm R2 = " << sqrt(int2d(disk)( u^2 ) )<< endl;
   41 : cout << "error L2 = " << sqrt(int2d(disk)( (u-(1-x^2-y^2)/4)^2 )) << endl;
   42 : cout << "d-norm D10 = " << sqrt(int2d(disk)( (dx(u))^2 )
   43 :                               + int2d(disk)( (dy(u))^2 ) )<< endl;
   44 : cout << "error H10 = " << sqrt(int2d(disk)( (dx(u)+x/2)^2 )
   45 :                               + int2d(disk)( (dy(u)+y/2)^2 ) )<< endl;
   46 : end sizestack + 1024 =2584  ( 1560 )

  --  mesh:  Nb of Triangles =   6960, Nb of Vertices 3581
  SkyLineMatrix: size pL/pU: 3581 218547 218547 moy=61.0296
  -- Solve :
          min 1.49925e-32  max 0.250008
u-norm R2 = 0.255724
error L2 = 0.000131351
d-norm D10 = 0.626483
error H10 = 0.00940587
times: compile 0.104s, execution 3.196s,  mpirank:0
 CodeAlloc : nb ptr  1069,  size :126336 mpirank: 0
Ok: Normal End
続行するには何かキーを押してください . . .
```