KnotTheory` Make.nb

This notebook generates KnotTheory/init.m from the modules in src/ and KnotTheory.zip and KnotTheory.tar.gz from their sources.

In the future this notebook will supersede the "makefile"; currently the makefile is not operational, but it is kept in the repository because it contains the list of files that go into DTCodes4Knots12To16.tar.gz and DTCodes4Knots12To16.zip which are not yet made by this notebook.

This notebook also contains some testing code.

The Make Part

Change and execute the following cell to set the path for "trunk" on your system:

```
Switch["Scott",
 "Dror", SetDirectory["C:/drorbn/projects/KnotTheory/svn/trunk"],
 "Scott", SetDirectory["~/projects/KnotTheory/trunk"],
 "Radmila", SetDirectory["C:\\radmila\\KnotTheory\\trunk"]
]
```

```
/Users/scott/projects/KnotTheory/trunk
```

Most times you will not need to edit the following variables, so just execute the following cell to initialize them:

```
PathOf["7z"] := "c:\\Program Files\\7-Zip";
InitSources = ("src/" <> #) & /@ {
    "Base.m", "Braids.m", "TubePlot.m", "DrawPD.m", "Data.m", "BraidData.m",
    "Naming.m", "GaussCode.m", "GC2PD.m", "Indiana.m", "HyperbolicVolume.m",
    "WikiForm.m", "HOMFLYPT.m", "Kauffman.m", "Kh.m", "MorseLink.m",
    "DrawMorseLink.m", "ML2PD.m", "AlexanderConway.m", "VogelsAlgorithm.m",
    "MultivariableAlexander.m", "REngine.m", "TestRMatrix.m", "CJREngine.m",
    "ColouredJones.m", "HFGenus.m", "KTtoLinKnot.m", "ArcPresentation.m", "HFK.m"
   };
DistributionFiles = Join[
   FileNames["*.m", "KnotTheory"],
   Select[
    FileNames["*", "KnotTheory" <> $PathnameSeparator <> #] & /@ {"JavaKh-v1", "JavaKh-v2",
       "WikiLink", "KnotAtlas", "scripts", "QuantumGroups", "HFK-Zurich"}, Infinity],
    (FileType[#] == File) && StringFreeQ[#, $PathnameSeparator <> "."] &
   ]
  ];
```

```
CreateZip[] := Module[{},
  DeleteFile["KnotTheory.zip"];
  Switch[$OperatingSystem,
    "Windows", Run[StringJoin[
      "\"" <> PathOf["7z"] <> "\\7z.exe\" ",
      "a -tzip KnotTheory.zip @distribution_files"
    ]],
    "MaxOSX" | _, Run["zip KnotTheory.zip `cat distribution_files`"]
  ];
]

CreateGZ[] :=
 Module[{},
  DeleteFile["KnotTheory.tar.gz"];
  Switch[$OperatingSystem,
    "Windows", Block[{},
    Run[StringJoin[
      "\"" <> PathOf["7z"] <> "\\7z.exe\" ",
      "a -ttar KnotTheory.tar @distribution_files"
    ]];
    Run["\"" <> PathOf["7z"] <> "\\7z.exe\" a -tgzip KnotTheory.tar.gz KnotTheory.tar"];
    DeleteFile["KnotTheory.tar"];
    ],
    "MacOSX" | _, Run["tar czvf KnotTheory.tar.gz `cat distribution_files`"]
  ];
 ]

f = OpenWrite["distribution_files", PageWidth → Infinity];
WriteString[f, StringJoin[# <> "\n" & /@ DistributionFiles]];
Close[f];
```

And now the action itself:

```
lines = StringReplace[#,
    "---date---" → ToString[Date[]]
  ] & /@ Import["src/System.mm", "Lines"];
lines = Flatten[{
    lines,
    {
      "(* Begin source file " <> # <> "*)\n",
      Import[#, "Lines"],
      "(* End source file " <> # <> "*)\n\n"
    } & /@ InitSources
  }];
Export["KnotTheory/init.m", lines, "Lines"];
CreateZip[];
CreateGZ[];

Directory[]
```

/Users/scott/projects/KnotTheory/trunk

The Test Part - Computations

```
SetAttributes[Assert, HoldAll];
Assert[fact_] := Module[
   {result = fact},
   If[result === True,
    True,
    Print[
     StringTake[ToString[Hold[fact]], {6, -2}],
     " yields ", fact]
   ]
  ];
```

<< KnotTheory`

Loading KnotTheory` version of September 6, 2014, 13:37:37.2841.
Read more at http://katlas.org/wiki/KnotTheory.

Assert[Jones[Link["L11n458"]][q] == Jones[DTCode[Link["L11n458"]]][q]]

KnotTheory::loading : Loading precomputed data in Jones4Links`.

KnotTheory::loading : Loading precomputed data in PD4Links`.

KnotTheory::credits : The GaussCode to PD conversion was written by Siddarth Sankaran at the University of Toronto in the summer
of 2005.

True

Assert$\left[Kh[PD[Knot[3, 1]]][q, t] == \frac{1}{q^3} + \frac{1}{q} + \frac{1}{q^9 t^3} + \frac{1}{q^5 t^2} \right]$

KnotTheory::loading : Loading precomputed data in PD4Knots`.

KnotTheory::credits : The Khovanov homology program JavaKh–v2 is an update of Jeremy Green's program JavaKh–v1, written by
Scott Morrison in 2008 at Microsoft Station Q.

Something went wrong running JavaKh; nothing was returned. The command line was:

```
!java -classpath
  "C:\drorbn\projects\KnotTheory\svn\trunk\KnotTheory\JavaKh-v2\bin;C:\drorbn\projects\KnotTheory
  \svn\trunk\KnotTheory\JavaKh-v2\jars\commons-cli-1.0.jar;C:\drorbn\projects\KnotTheory\svn\
  trunk\KnotTheory\JavaKh-v2\jars\commons-io-1.2.jar;C:\drorbn\projects\KnotTheory\svn\trunk\
  KnotTheory\JavaKh-v2\jars\commons-logging-1.1.jar;C:\drorbn\projects\KnotTheory\svn\trunk\
  KnotTheory\JavaKh-v2\jars\log4j-1.2.12.jar"
  org.katlas.JavaKh.JavaKh --mod 0 < pd 2> JavaKh.log
```

There may have been an error log produced by Java:

```
'java' is not recognized as an internal or external command,
operable program or batch file.
```

```
                              1    1     1        1
Hold[Kh[PD[Knot[3, 1]]][q, t] == -- + - + ----- + -----]
                              3    q     9  3     5  2
                             q          q  t     q
            1    1    1      1
  yields $Failed[q, t] == -- + - + ----- + -----
            q³   q   q⁹ t³   q⁵ t²
```

$$\text{Assert}\big[\text{Kh}[\text{PD}[\text{Knot}[4, 1]]], \text{Program} \to \text{"FastKh"}][q, t] == \frac{1}{q} + q + \frac{1}{q^5 t^2} + \frac{1}{q t} + q t + q^5 t^2\big]$$

KnotTheory::credits : The Khovanov homology program FastKh was written by Dror Bar–Natan.

True

$$\text{Assert}\big[\text{Last}[\text{Kh}[\text{TorusKnot}[6, 5], \text{Modulus} \to \text{Null}][q, t]] == q^{39} t^{12} \text{ZMod}[2, 5]\big]$$

Something went wrong running JavaKh; nothing was returned. The command line was:

```
!java -classpath
  "C:\drorbn\projects\KnotTheory\svn\trunk\KnotTheory\JavaKh-v2\bin;C:\drorbn\projects\KnotTheory
  \svn\trunk\KnotTheory\JavaKh-v2\jars\commons-cli-1.0.jar;C:\drorbn\projects\KnotTheory\svn\
  trunk\KnotTheory\JavaKh-v2\jars\commons-io-1.2.jar;C:\drorbn\projects\KnotTheory\svn\trunk\
  KnotTheory\JavaKh-v2\jars\commons-logging-1.1.jar;C:\drorbn\projects\KnotTheory\svn\trunk\
  KnotTheory\JavaKh-v2\jars\log4j-1.2.12.jar"  org.katlas.JavaKh.JavaKh -Z < pd 2> JavaKh.log
```

There may have been an error log produced by Java:

```
'java' is not recognized as an internal or external command,
operable program or batch file.
```

$$\text{Hold}[\text{Last}[\text{Kh}[\text{TorusKnot}[6, 5], \text{Modulus} \to \text{Null}][q, t]] == q^{39} \ t^{12} \ \text{ZMod}[2, 5]$$
$$\text{yields } t == q^{39} \ t^{12} \ \text{ZMod}[2, 5]$$

$$\text{Assert}\big[\text{Alexander}[\text{"K11a44"}, 2][t] === \{1 - t + t^2\}\big]$$

KnotTheory::loading : Loading precomputed data in DTCode4KnotsTo11`.

KnotTheory::credits : The program Alexander[K, r] to compute Alexander ideals was written by Jana Archibald at the University of Toronto in the summer of 2005.

True

```
Assert[
 Alexander["10_99", #][t] & /@ {1, 2} ===
```
$$\{\{1 - 4 t + 10 t^2 - 16 t^3 + 19 t^4 - 16 t^5 + 10 t^6 - 4 t^7 + t^8\}, \{1 - 2 t + 3 t^2 - 2 t^3 + t^4\}\}$$
```
]
```

True

```
Assert[
 ArcPresentation["K11n11"] === ArcPresentation[{12, 2}, {1, 10}, {3, 9},
   {5, 11}, {9, 12}, {4, 8}, {2, 5}, {11, 7}, {8, 6}, {7, 4}, {10, 3}, {6, 1}]
]
```

KnotTheory::credits : MorseLink was added to KnotTheory` by Siddarth Sankaran

at the University of Toronto in the summer of 2005.

True

```
Assert[
 BR[ArcPresentation[
    {12, 2}, {1, 10}, {3, 9}, {5, 11},
    {9, 12}, {4, 8}, {2, 5}, {11, 7}, {8, 6}, {7, 4}, {10, 3}, {6, 1}]
  ] === BR[4, {-1, 2, 3, 3, -2, -1, -2, -1, 2, 3, 3, 3, 2}]
]
```

KnotTheory::credits : Vogel's algorithm was implemented by Dan Carney in the summer of 2005 at the University of Toronto.

True

$$\texttt{Assert}\Big[(\texttt{HFKHat}[\texttt{Knot}[8, 19]][t, m] \texttt{ /. } m \to 1/t) == \frac{4}{t^3} + \frac{1}{t^2}\Big]$$

KnotTheory::loading : Loading precomputed data in HFKHat4KnotsTo11`.

True

$$\texttt{Assert}\Big[(\texttt{HFKHat}[\texttt{BR}[\texttt{Knot}[8, 19]]][t, m] \texttt{ /. } m \to 1/t) == \frac{4}{t^3} + \frac{1}{t^2}\Big]$$

KnotTheory::credits : The minimum braids representing the knots with up to 10 crossings were provided by Thomas Gittings. See arXiv:math.GT/0401051.

KnotTheory::credits : The HFKHat program was written by Jean–Marie Droz in 2007 at the University of Zurich, based on methods of Anna Beliakova's arXiv:07050669.

$$\texttt{Hold}[(\texttt{HFKHat}[\texttt{BR}[\texttt{Knot}[8, 19]]][t, m] \texttt{ /. } m \to \frac{1}{t}) == \frac{1}{t^3} + \frac{4}{3} + \frac{1}{2}\texttt{]}$$
$$t \quad \texttt{yields \$Failed}\Big[t, \frac{1}{t}\Big] == \frac{4}{t^3} + \frac{1}{t^2}$$

$$\texttt{Assert}\Big[(\texttt{Plus @@ }(\texttt{AlternatingQ /@ AllKnots}[8])) == 3\,\texttt{False} + 18\,\texttt{True}\Big]$$

True

```
Assert[AlternatingQ[Knot[0, 1]] == True]
```

True

```
Assert[
 MultivariableAlexander[Link["L7a7"]][t] == MultivariableAlexander[PD[Link["L7a7"]]][t]
]
```

KnotTheory::loading : Loading precomputed data in MultivariableAlexander4Links`.

KnotTheory::credits : The multivariable Alexander program "MVA2" was written by Jana Archibald at the University of Toronto in 2007–2008.

True

```
Flip[X[i_, j_, k_, l_]] := If[l == j + 1 || j - l > 1, X[j, k, l, i], X[l, i, j, k]];
VassilievCube[{}, pdcomb_] := pdcomb;
VassilievCube[{i_, rest___}, pdcomb_] := VassilievCube[{rest},
  Expand[pdcomb - (pdcomb /. pd_PD :> MapAt[Flip, pd, i])]]
 ]
```

```
Assert[Equal[
  Series[
      VassilievCube[{1, 2, 5, 7}, PD[#]]
        /. pd_PD :> MultivariableAlexander[pd][t] /. t[i_] → E^(h x[i]),
      {h, 0, 2}
    ] & /@ AllLinks[7],
  {O[h]³, O[h]³, O[h]³, O[h]³, O[h]³, O[h]³, O[h]³, O[h]³, O[h]³}
]]
```

True

```
Assert[Equal[
  Series[
    VassilievCube[{1, 2, 5, 7}, PD[Link[7, Alternating, 1]]]
      /. pd_PD :> MultivariableAlexander[pd, Program → "MVA1"][t] /.
    t[i_] → E^(h x[i]),
    {h, 0, 2}
  ],
  -2 x[1] h + (-2 x[1]² - x[1] x[2]) h² + O[h]³
]]
```
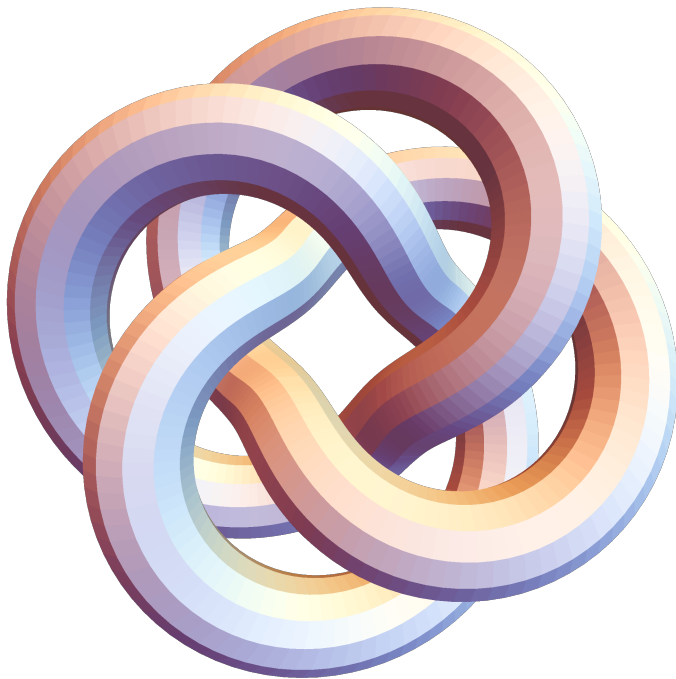
KnotTheory::credits : The multivariable Alexander program "MVA1" was written by Dan Carney at the University of Toronto in the
                      summer of 2005.

True

The Test Part - Graphics

```
TubePlot[TorusKnot[4, 3]] // Show
```

```
Draw[ArcPresentation[Knot[9, 17]]]
```