

TP C++

Phase 2

Liste des fonctionnalités :

- Fonctions simples : cos, sin, tan, sqrt, pow, log, exp, lerp, hypot.
- Fonction variadique : polynome.

Les fonctions sont gérées grâce à des lambdas, pour en ajouter il suffit de modifier *ProgFunction::getFunctionLambda()*.

Le nombre d'arguments d'une fonction est vérifié à l'évaluation de la fonction.

Choix:

- J'ai continué à partir de la correction fournie, mon traitement de la phase 1 n'étant pas suffisamment général pour supporter facilement les modifications.
- Les fonctions sont des lambdas prenant en entrée un tableau d'arguments. Les arguments sont évalué grâce à Programme, cela permet de facilement gérer des arguments tels que "a+b" ou "pow(x,y)". Une conséquence est qu'un programme, par exemple "y = 2;pow(2,y)", est un argument valide (note : "(y = 2;pow(2,y))" n'est pas un argument valide, car ce n'est pas un programme valide).
- Ma façon de gérer les arguments des fonctions (un tableau) me permettrait de me passer du 1er arguments de polynome, mais pour coller au sujet je l'ai rajouté.

Compilation :

- mkdir build && cd build
- cmake ..
- make

Exécution :

- calculator : entrées utilisateur.
- calculator <numTest> : exécute le test, il y a 8 tests numérotés de 0 à 7.