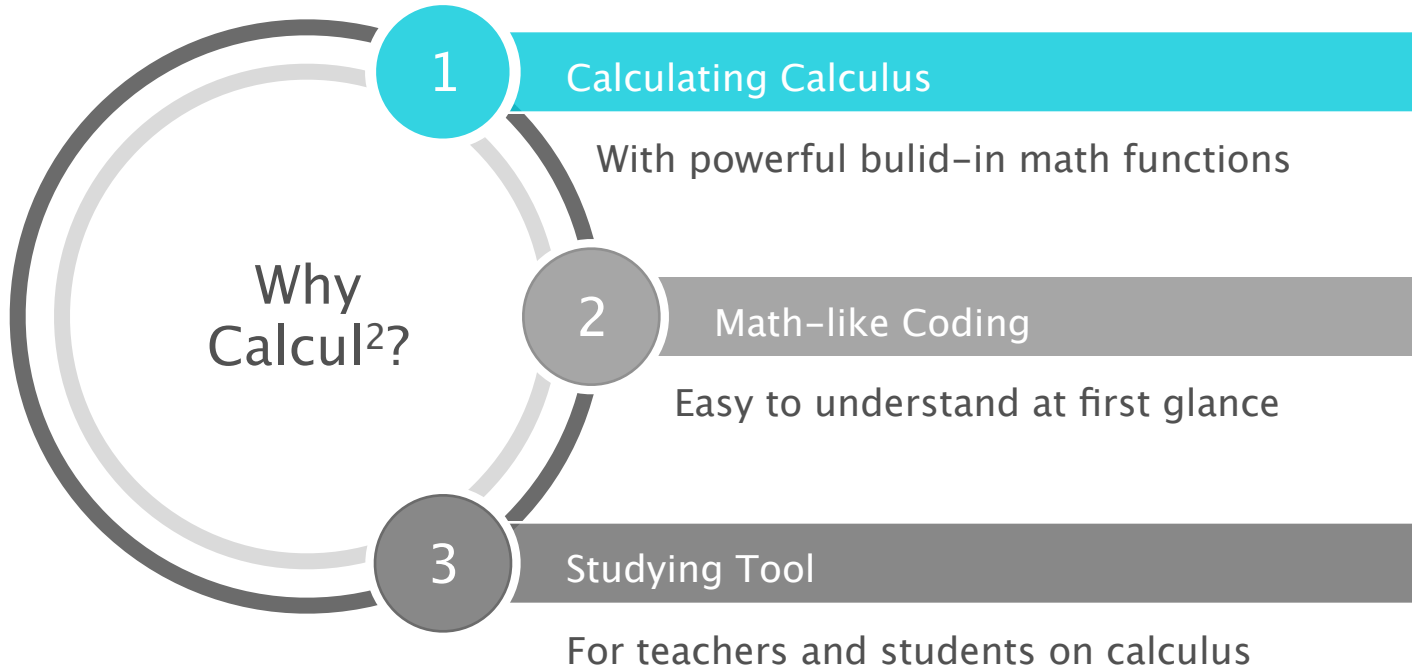


COMS W4115 Project

Calcul²

»» Final Report

1. Simple calculus calculating language for mathematical function evaluation, derivation and integration.
2. Build-in types: Floats, Single and Multivariate Math Functions.
3. Dynamic and Strong typed.



Function Declaration: $f(x)=x^2$;

Function Evaluation: $f(3)$;

Function Derivation: $f'(x)$;

Function Integral: $\int f(x)$;

Multivariate Functions:

Declaration: $g(x,y)=x+y^2$;

Function Evaluation: $g(1,2)$;

Function Derivation: $g'(x)$;

Mathematical Functions

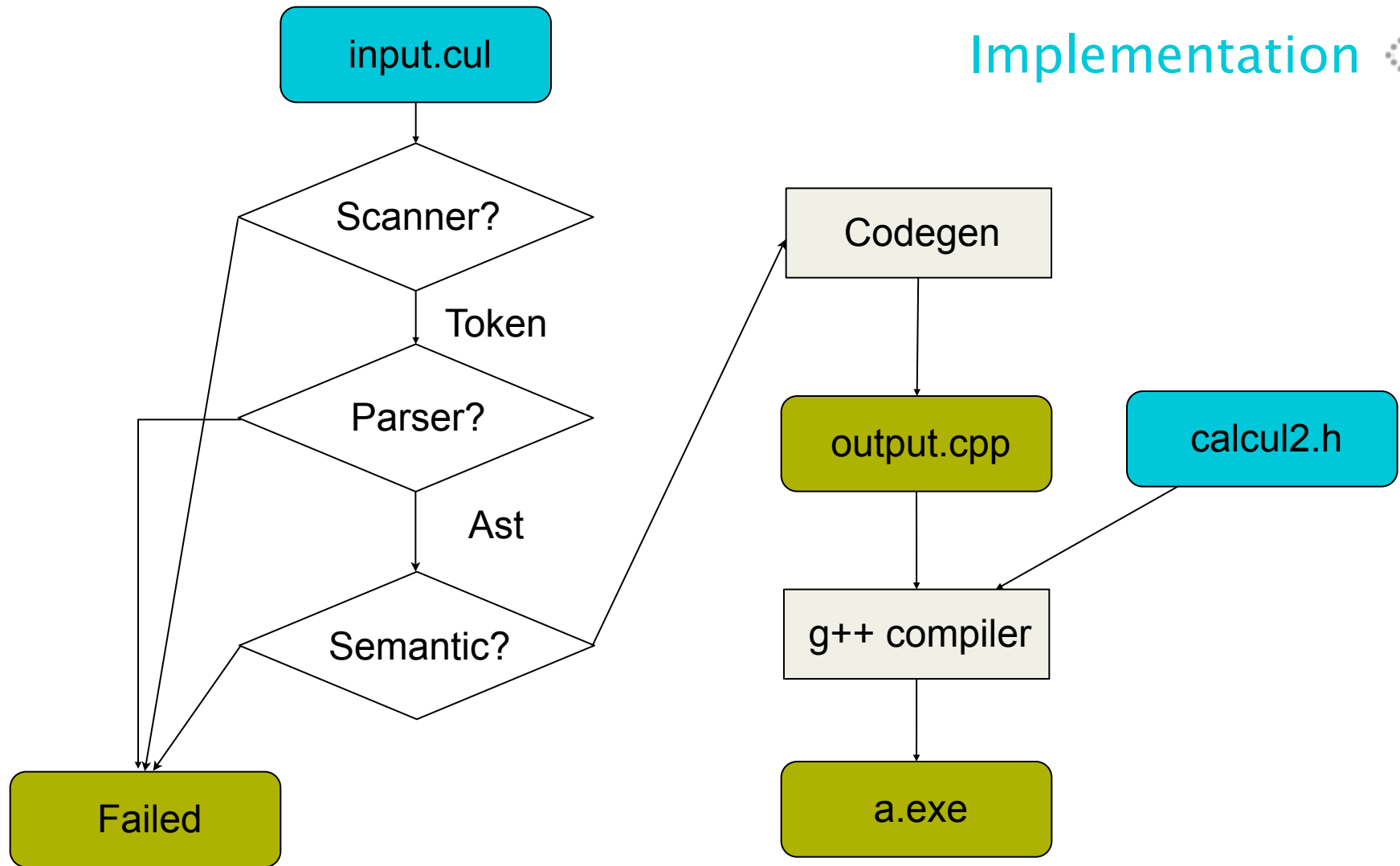
input.cul

```
main()
{
  f($x) = 2 * x;
  g($x) = sin(x);

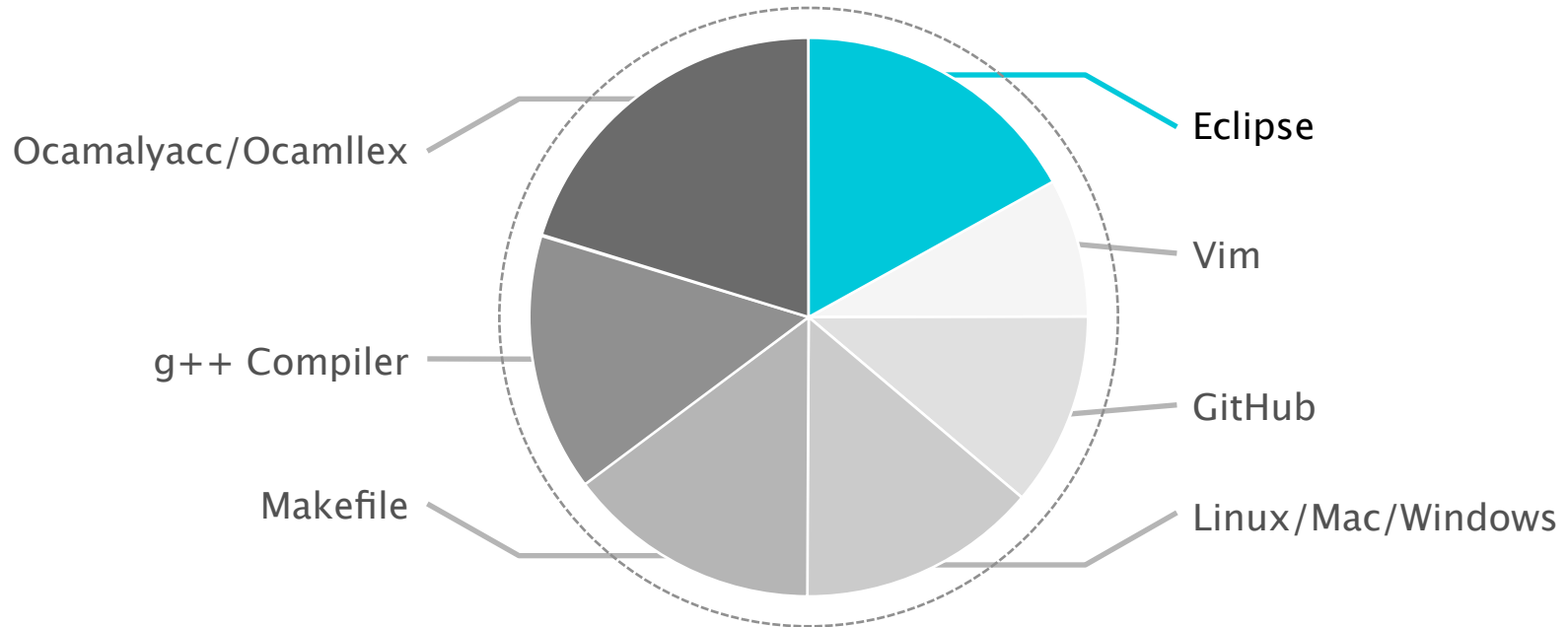
  h($x) = f + g;
  :h;
  :h'(x);
}
```

output

```
2 * x + sin(x)
2 + cos(x)
```



Development Tools/ Environment



input.cul

```
main()
{
  f($x) = x ^ 3 + 3 * x;
  :f'(x);
}
```

```
int main()
{
  double printer;

  vector<string> f_var;
  f_var.push_back("x");
  vector<double> f_begin, f_end, f_now;
  FTree f(f_var);

  f.AddNode(new FNode(T_OP,0,PLUS));
  f.AddNode(new FNode(T_OP,0,POWER));
  f.AddNode(new FNode(T_VAR,0,0));
  f.AddNode(new FNode(T_VAL,3.));
  f.AddNode(new FNode(T_OP,0,TIMES));
  f.AddNode(new FNode(T_VAL,3.));
  f.AddNode(new FNode(T_VAR,0,0));

  f_now.clear();
  f.Derive("x") -> Print();
  cout << "\n";

  return 0;
}
```

output

$$3 * x ^ 2 + 3$$

Project Lives on Teamwork

Plan Limited by Time

Problems Lessen by Testing

Practical Lively Things

THANKS!

