Algorithm Analysis, Assignment 1

Jacob Anabi

2021 February 25

1 Assignment Part 1:

Consider the 32 bit addition algorithm we discussed in class:

```
uint32_t addto32(uint32_t as[], int sz_a, uint32_t bs[], int sz_b) {
    // Assume that sz_b <= sz_a
    // Compute as += bs
    int i;
    uint32_t c = 0;
    uint64_t s;
    for (i=0; i< sz_b; i++) {
        s = (uint64_t) as[i] + (uint64_t) bs[i] + (uint64_t) c; // s is a 33 bit value
        c = s >> 32;
        as[i] = (uint32_t) s;
    }
    for ( ; i< sz_a; i++) {
        s = (uint64_t) as[i] + (uint64_t) c; // s is a 33 bit value
        c = s >> 32;
        as[i] = (uint32_t) s;
    }
    return c;
}
```

We want to prove that

$$\mathrm{c} + \sum_{k<i} \mathrm{as[k]} = \sum_{k<i} (\mathrm{as'[k]} + \mathrm{bs[k]})$$

where as' denotes the original value of as.

We will do so by induction on i.

Proof.

Inductive Basis ($i = 0$):

If $i = 0$ then

$$\begin{aligned} \text{c} + \sum_{k<0} \text{as[k]} &= \text{c} + 0 \\ &= 0 + 0 \text{ [since c is initialized to 0]} \\ &= 0 \\ &= \sum_{k<0} (\text{as'[k] + bs[k]}) \end{aligned}$$

Thus, the equality holds for our basis.

Inductive Hypothesis ($i = n$):

Suppose

$$\text{c} + \sum_{k<n} \text{as[k]} = \sum_{k<n} (\text{as'[k] + bs[k]})$$

Inductive Step ($i = n + 1$):

$$\begin{aligned} \sum_{k<n+1} (\text{as'[k] + bs[k]}) &= \sum_{k<n} (\text{as'[k] + bs[k]}) + \text{as'[n] + bs[n]} \\ &= \text{c} + \sum_{k<n} \text{as[k]} + (\text{as'[n] + bs[n]}), \text{ by the Inductive Hypothesis} \\ &= \sum_{k<n} \text{as[k]} + (\text{c} + \text{as'[n] + bs[n]}) \\ &= \sum_{k<n} \text{as[k]} + \text{s}, \text{ by the assignment of s in the for loop} \\ &= \sum_{k<n} \text{as[k]} + \text{(uint32\_t) s} + \text{s >> 32} \\ &= \sum_{k<n} \text{as[k]} + \text{a[n]} + \text{c}, \text{ by the assignments in the for loop} \\ &= \text{c} + \sum_{k<n+1} \text{as[k]} \end{aligned}$$

$\square$

2 Assignment Part 2:

Consider the 32 bit partial product algorithm:

```
void partialprod32
(uint32_t as[], int sz_a, uint32_t bs[], int sz_b, uint32_t d, int shift)
{
    // Assume sz_b + sz_c <= sz_a => sz_b < sz_a
    // (this is because if we have two numbers with n and m digits,
    // then their product would be n + m digits at most)
    // Compute as += bs * d

    int i;
    int i_shifted;
    uint32_t p_split = 0;
    uint32_t c = 0;
    uint64_t s;
    uint64_t p;

    for (i = 0; i < sz_b; i++)
    {
        i_shifted = i+shift;
        p = (uint64_t) bs[i] * (uint64_t) d; // p is a 64 bit value
        s = (uint64_t) as[i_shifted] + (uint64_t) ((uint32_t) p)
          + (uint64_t) p_split + (uint64_t) c;
        p_split = p >> 32;
        c = s >> 32;
        as[i_shifted] = (uint32_t) s;
    }

    for ( ; i < sz_a-shift; i++)
    {
        if (p_split == 0 && c == 0)
        {
            break;
        }
        i_shifted = i+shift;
        s = (uint64_t) as[i_shifted] + (uint64_t) p_split + (uint64_t) c;
        p_split = 0;
        c = s >> 32;
        as[i+shift] = (uint32_t) s;
    }
}
```

We will prove, by induction on i, that

$$\text{c} + \text{p\_split} + \sum_{k<i} \text{as[k+w]} = \sum_{k<i} (\text{as'[k+w]} + \text{bs[k+w]} * \text{d})$$

where as' denotes the original value of as and w $\in \mathbb{N}$.

Proof.
Inductive Basis ($i = 0$):
If $i = 0$ then

$$\begin{aligned}
\text{c} + \text{p_split} + \sum_{k<0} \text{as[k+w]} &= \text{c} + \text{p_split} + 0 \\
&= 0 + 0 + 0 \text{ [since c and p_split are initialized to 0]} \\
&= 0 \\
&= \sum_{k<0} (\text{as'[k+w]} + \text{bs[k]} * \text{d})
\end{aligned}$$

Thus, the equality holds for our basis.

Inductive Hypothesis ($i = n$):
Suppose

$$\text{c} + \text{p\_split} + \sum_{k<n} \text{as[k+w]} = \sum_{k<n} (\text{as'[k+w]} + \text{bs[k]} * \text{d})$$

Inductive Step ($i = n+1$):

$$\sum_{k<n+1} \text{(as'[k+w] + bs[k] * d)}$$
$$= \sum_{k<n} \text{(as'[k+w] + bs[k] * d)} + \text{as'[n+w] + bs[n] * d}$$
$$= \text{c} + \text{p\_split} + \sum_{k<n} \text{as[k+w]} + \text{(as'[n+w] + bs[n] * d)}, \text{ by the I.H.}$$
$$= \sum_{k<n} \text{as[k+w]} + \text{(c} + \text{p\_split} + \text{as'[n+w] + p)}$$
$$= \sum_{k<n} \text{as[k+w]} + \text{(c} + \text{p\_split} + \text{as'[n+w] + (uint64\_t) ((uint32\_t) p) + p >> 32)}$$
$$= \sum_{k<n} \text{as[k+w]} + \text{(c} + \text{p\_split} + \text{as'[n+w] + (uint64\_t) ((uint32\_t) p) + p\_split')}$$
$$= \text{p\_split'} + \sum_{k<n} \text{as[k+w]} + \text{(c} + \text{p\_split} + \text{as'[n+w] + (uint64\_t) ((uint32\_t) p))}$$
$$= \text{p\_split'} + \sum_{k<n} \text{as[k+w]} + \text{s}$$
$$= \text{p\_split'} + \sum_{k<n} \text{as[k+w]} + \text{(uint32\_t) s} + \text{s >> 32}$$
$$= \text{p\_split'} + \sum_{k<n} \text{as[k+w]} + \text{as[n+w]} + \text{s >> 32}$$
$$= \text{p\_split'} + \sum_{k<n} \text{as[k+w]} + \text{as[n+w]} + \text{c}$$
$$= \text{c} + \text{p\_split'} + \sum_{k<n} \text{as[k+w]} + \text{as[n+w]}$$
$$= \text{c} + \text{p\_split'} + \sum_{k<n+1} \text{as[k+w]}$$
$$= \text{c} + \text{p\_split} + \sum_{k<n+1} \text{as[k+w]}, \text{ by renaming p\_split' to p\_split}$$

$\square$

3 Assignment Part 3:

Consider the 64 bit multiplication algorithm:

```
void bigmul64(uint64_t a[], int sz_a, uint64_t b[], int sz_b, uint64_t c[], int sz_c)
{
    uint32_t *as = (uint32_t *) a;
    uint32_t *bs = (uint32_t *) b;
    uint32_t *cs = (uint32_t *) c;

    int i;

    for (i = 0; i < 2*sz_c; i++)
    {
        partialprod32(as, 2*sz_a, bs, 2*sz_b, cs[i], i);
    }
}
```

We will prove that

$$\sum_{w<m}(\text{c} + \text{p\_split} + \sum_{k<i}\text{as[k+w]}) = \sum_{w<m}\sum_{k<i}(\text{as'[k+w]} + \text{bs[k+w]} * \text{d})$$

where as' denotes the original value of as.

Proof.
Well from part 2, we know that:

$$\text{c} + \text{p\_split} + \sum_{k<i}\text{as[k+w]} = \sum_{k<i}(\text{as'[k+w]} + \text{bs[k+w]} * \text{d})$$

Thus,

$$\sum_{w<m}(\text{c} + \text{p\_split} + \sum_{k<i}\text{as[k+w]}) = \sum_{w<m}\sum_{k<i}(\text{as'[k+w]} + \text{bs[k+w]} * \text{d})$$

$\square$